Defense Information Systems Agency
Center for Standards

DEPARTMENT OF DEFENSE TECHNICAL ARCHITECTURE FRAMEWORK FOR INFORMATION MANAGEMENT

Volume 6: Department of Defense (DoD) Goal Security Architecture

Version 3.0

30 April 1996

FOREWORD:
ABOUT THIS DOCUMENT

This edition of the Technical Architecture Framework for Information Management (TAFIM) replaces Version 2.0, dated 30 June 1994. Version 3.0 comprises eight volumes, as listed on the following configuration management page.

TAFIM HARMONIZATION AND ALIGNMENT

This TAFIM version is the result of a review and comment coordination period that began with the release of the 30 September 1995 Version 3.0 Draft. During this coordination period, a number of extremely significant activities were initiated by DoD. As a result, the version of the TAFIM that was valid at the beginning of the coordination period is now "out of step" with the direction and preliminary outcomes of these DoD activities. Work on a complete TAFIM update is underway to reflect the policy, guidance, and recommendations coming from theses activities as they near completion. Each TAFIM volume will be released as it is updated. Specifically, the next TAFIM release will fully reflect decisions stemming from the following:

- The DoD 5000 Series of acquisition policy and procedure documents
- The Joint Technical Architecture (JTA), currently a preliminary draft document under review.
- The C4ISR Integrated Task Force (ITF) recommendations on Operational, Systems, and Technical architectures.

SUMMARY OF MAJOR CHANGES AND EXPECTED UPDATES

This volume, Volume 6 of the TAFIM, has been changed from the previous edition to place a greater emphasis on the specific phases of the system engineering process, and how each feeds into the next. A significant attempt has been made to impose a consistent story-line on abstract and generic architecture views and security allocations for all elements. Additionally, some restructuring of the volume was done to make navigation through the document flow more consistently and coherently. Information pertaining to standards appearing in this volume has been updated to reflect current situations.

The next edition of this volume will be updated as necessary to reflect the DoD policies changes and decisions noted above.

A NOTE ON VERSION NUMBERING

A version numbering scheme approved by the Architecture Methodology Working Group will control the version numbers applied to all future editions of TAFIM volumes. Version numbers will be applied and incremented as follows:

- This edition of the TAFIM is the official Version 3.0.
- From this point forward, single volumes will be updated and republished as needed. The second digit in the version number will be incremented each time (e.g., Volume 7 Version 3.1). The new version number will be applied only to the volume(s) that are updated at that time. There is no limit to the number of times the second digit can be changed to account for new editions of particular volumes.
- On an infrequent basis (e.g., every two years or more), the entire TAFIM set will be republished at once. Only when all volumes are released simultaneously will the first digit in the version number be changed. The next complete version will be designated Version 4.0.
- TAFIM volumes bearing a two-digit version number (e.g., Version 3.0, 3.1, etc.) without the DRAFT designation are final, official versions of the TAFIM. Only the TAFIM program manager can change the two-digit version number on a volume.
- A third digit can be added to the version number as needed to control working drafts, proposed volumes, internal review drafts, and other unofficial releases. The sponsoring organization can append and change this digit as desired.

Certain TAFIM volumes developed for purposes outside the TAFIM may appear under a different title and with a different version number from those specified in the configuration management page. These editions are not official releases of TAFIM volumes.

DISTRIBUTION

Version 3.0 is available for download from the DISA Information Technology Standards Information (ITSI) bulletin board system (BBS). Users are welcome to add the TAFIM files to individual organizations' BBSs or file servers to facilitate wider availability.

This final release of Version 3.0 will be made available on the World Wide Web (WWW) shortly after hard-copy publication. The Defense Information Systems Agency (DISA) is also investigating other electronic distribution approaches to facilitate access to the TAFIM and to enhance its usability.

TAFIM Document Configuration Management Page

The latest **authorized versions of the TAFIM** volumes are as follows:

Volume 1:	Overview	3.0	30 April 1996
Volume 2:	Technical Reference Model	3.0	30 April 1996
Volume 3:	Architecture Concepts & Design Guidance	3.0	30 April 1996
Volume 4:	DoD SBA Planning Guide	3.0	30 April 1996
Volume 5:	Program Manager's Guide for Open Systems	3.0	30 April 1996
Volume 6:	DoD Goal Security Architecture	3.0	30 April 1996
Volume 7:	Adopted Information Technology Standards	3.0	30 April 1996
Volume 8:	HCI Style Guide	3.0	30 April 1996

Other working drafts may have been released by volume sponsors for internal coordination purposes. It is not necessary for the general reader to obtain and incorporate these unofficial, working drafts.

Note: Only those versions listed above as authorized versions represent official editions of the TAFIM.

This page intentionally left blank.

CONTENTS

FIGURES

1.0 INTRODUCTION

The Defense Information Systems Security Program (DISSP) was initiated at the request of the Assistant Secretary of Defense (Command, Control, Communications and Intelligence). The Defense Information Systems Agency (DISA) and the National Security Agency (NSA) agreed to cooperate in achieving eight security objectives. These objectives were in the areas of:

- Security policy
- Architecture
- Standards and protocols
- Accreditation procedures
- Technology
- Transition planning
- Organizational improvement
- Products and services availability.

Accordingly, a DISSP Office was established and among its responsibilities was the development of the Department of Defense (DoD) Goal Security Architecture (DGSA). The DISSP has since become a part of the CISS in DISA. The Center for Information System Security (CISS) assists DoD organizations in the transition of existing systems and in the development of new systems in accordance with the DGSA.

Concurrent with the development of the DGSA, efforts were underway within DISA to define information system architectures for the Defense Information System (DIS). These efforts focused on the Technical Architecture Framework for Information Management (TAFIM). The TAFIM is intended to be generic and sufficiently flexible in its definition so that specific systems may be developed or modified to satisfy specific mission goals. The TAFIM is thereby a "goal information system architecture" and has incorporated the DGSA, as Volume 6, as its "goal security architecture."

1.1 PURPOSE

The DGSA was developed in conjunction and harmony with the total requirements for automated services. The protection of information and system assets was a key consideration as part of the total view of objectives, threats, performance, interoperability, extensibility, usability, and cost of implementations. The DGSA does not provide a specification for any particular information system or component. Rather, it specifies security principles and target security capabilities that will guide system security architects in creating specific security architectures

that are consistent with the DGSA. While there is no fixed date by which all aspects of the DGSA will be achieved, the concepts of the DGSA can be applied to information systems today. As security technology improves and products incorporate support for DGSA concepts, specific information systems will achieve greater and greater consistency with their individual goals.

After the initial release of the DGSA, activities were undertaken to create a DGSA Transition Plan (CISS, 1995) to define the steps needed to incorporate DGSA concepts into information systems. The Transition Plan is intended for system planners and managers addressing security in information system development or modernization programs. It may also be used by commercial developers, vendors, and those interested in incorporating specific security initiatives or objectives outlined in the Transition Plan into their product developments or security programs. System security engineers and integrators will be able to take advantage of the development of security products and mechanisms that will result from implementation of the Transition Plan. Like the DGSA, the Transition Plan is a living document that will be updated periodically to take into account changes in technology and new application areas.

1.2 SCOPE

The DISSP was instituted to draw together various information system applications, information transport systems, programs, and architectural activities to bring about consistency, efficiency, and interoperability in the security designs for the DIS. Several programs and systems were identified, such as the Defense Message System (DMS), the Defense Information Systems Network (DISN), the Integrated Tactical/Strategic Data Network (ITSDN), and the DoD Multilevel Security (MLS) Program, as well as emerging applications such as electronic commerce, as candidates from which DISSP personnel could gather a complete set of security requirements. These programs cover the bulk of the DIS and are reasonable representatives of DoD information processing needs as well as those of commercial and Federal communities. The DGSA encompasses this diversity of information systems to achieve greater efficiency and interoperability throughout the DIS and other communities.

1.3 ARCHITECTURAL TYPES

Information system architectures range in definition and occur in sequence from abstract views to specific views of what is to be developed. Experience shows that four types are frequently used: abstract, generic, logical, and specific. The TAFIM is considered to be an abstract and generic architecture and the DGSA, as part of the TAFIM, is also abstract and generic.

1.3.1 Abstract Architecture

An abstract architecture begins with knowledge of the requirements and defines corresponding functions to be performed. It defines principles and fundamental concepts that guide the selection and organization of functions. Abstract security architectures cite principles, fundamental concepts, and functions that satisfy the typical security requirements. These

concepts and functions are allocated to elements of an abstract definition of the information system architecture.

1.3.2 Generic Architecture

The development of a generic architecture is based upon the abstract architectural decisions. It defines the general types of components and allowable standards to be used, and identifies any necessary guidelines for their application. A generic security architecture proceeds from an initial allocation of security services and functions and begins to define the types of components and security mechanisms that are available to implement the security services with particular strengths. Any limitations in combining components and mechanisms because of incompatibility or security degradation must be cited in the guidelines for application.

1.3.3 Logical Architecture

A logical architecture is a design that meets a hypothetical set of requirements. It serves as a detailed example that illustrates the results of applying a generic architecture to specific circumstances. The only differences between a logical and a specific architecture are that the specific requirements are real, not hypothetical, and since the logical architecture is not intended to be implemented there is no need to perform a cost analysis. In logical security architectures, the logical design is accompanied by an illustration of the security analysis to be performed in specific architectures.

1.3.4 Specific Architecture

The objective of any system architect is to accomplish a level of design specification such that components may be acquired to implement the system. The specific architecture addresses components, interfaces, standards, performance, and cost. Specific security architectures show how all the selected information security components and mechanisms, including doctrine and supporting security management components, combine to meet the security requirements of the specific system under consideration.

1.4 DOCUMENT ORGANIZATION

Section 2 introduces the broad set of requirements to which the DGSA is responsive. The reflection of these requirements in a security policy and their use within a systems engineering process is discussed. In Section 3, an abstract information system architecture is presented, which includes the identification of major components of a generic information system; an abstract information model is discussed; and security responsibilities are allocated to the major architectural components based upon realistic expectations of the protections that can be achieved. Section 3 also presents several key security concepts used throughout the remainder of this document. The major components identified in Section 3 are then considered in detail, specifically end systems and relay systems in Section 4, security management in Section 5, transfer systems in Section 6, and administrative and environmental security measures in Section 7. Section 8 presents a logical architecture example of the application of the DGSA.

This page intentionally left blank.

2.0 SECURITY POLICY, REQUIREMENTS, AND ARCHITECTURES

This section first discusses the relationships between security policy and security requirements and how they are used within a systems engineering process to create a security architecture. Then, the security policy and security requirements upon which the DGSA are based are presented. Finally, some additional factors which influence security architecture choices are discussed.

2.1 SECURITY POLICY AND SECURITY REQUIREMENTS

Organizations often group their activities within one or more *missions* that focus on some subset of the organization's objectives. An *information system* is a collection of information processing and communications components, and the environment in which they operate, used to support the operations of one or more missions. A *security policy* pertains to organizations and their missions and is based upon the threats to mission accomplishment. A security policy (or, in a more general sense, a collection of security polices) documents the *security requirements* to be placed upon resources used by an organization. These security requirements express, for the organization's personnel, the organization's desired protection for its information and other system resources.

A security architecture designed to meet a specific mission's security requirements defines appropriate security services and mechanisms and allocates them to components of the mission's information system architecture. Since the DGSA is intended to address the needs of all DoD organizations, it is a more general statement about the common collection of services and mechanisms any information system might offer and allocates the security services and mechanisms to the generic components of an information system architecture.

Figure 2-1 shows that security policy and security requirements are derived as a result of examining the threats to a mission and are therefore a subset of the mission's requirements. It also indicates the strong relationship among mission, users, information, and policy. The DoD organizations that will employ the DGSA have many different missions. The security policy addressed by the DGSA is a general expression of the security requirements commonly found among the mission requirements of DoD organizations.

Security requirements are established in the same ways, whether for an entire organization or for a specific mission. The information to be managed is identified; the operational requirements for the use of the information are stated; the value of the information is determined; and the potential threats to the information are identified. Then, the security policy for either the entire organization or a specific mission can be stated in terms of the requirements for:

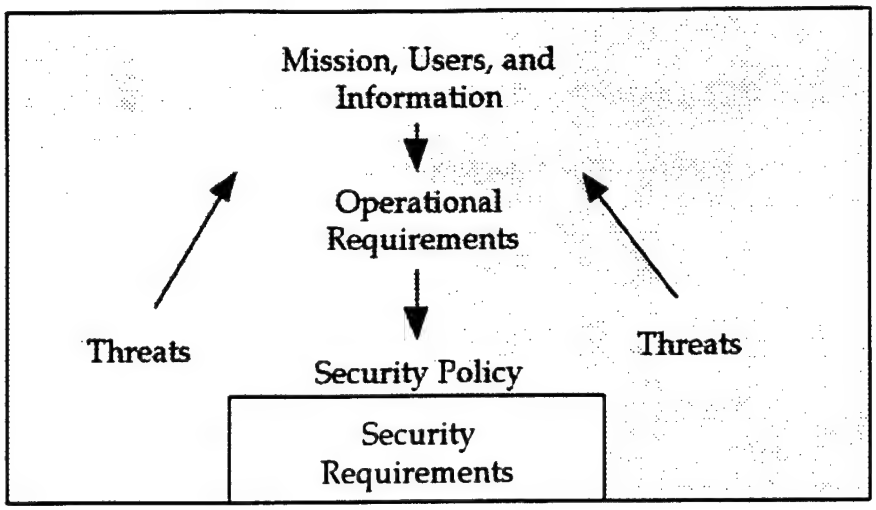

Figure 2-1. Derivation of Security Policy and Security Requirements

- Protection of the information based on the potential threats
- Security services that afford the appropriate protection of the information based upon the value of the information and the threats to it.

2.2 SECURITY ARCHITECTURE DEVELOPMENT

The development of security architectures, whether for entire organizations or for specific missions, are properly part of a larger systems engineering process. The process starts with a mission statement and progresses through a set of well-defined steps that culminate in the deployment and maintenance of information system components that satisfy organizational and mission needs. The first few steps of the process lead to an information system architecture that includes the security architecture. As a result, the security architecture, although separately identified, must be created in conjunction with the information system architecture. Mission-specific information system and security architectures are bounded by architectural decisions made in the higher level organizational architectures.

Figure 2-2 presents the first steps of a security engineering process showing the development of a mission-specific architecture and its relationship to a broader organizational architecture. Starting from a set of DoD requirements for the general DoD mission, a draft DoD Security Policy was created (see Section 2.3) and within the framework provided by the TAFIM, the DGSA was developed. Thus, the DGSA is responsive to the full range of DoD missions.

The development of a mission-specific security architecture begins by applying the DoD security policy to the specific mission requirements in order to develop a mission-specific security policy. The mission-specific security policy includes identification of the appropriate security services

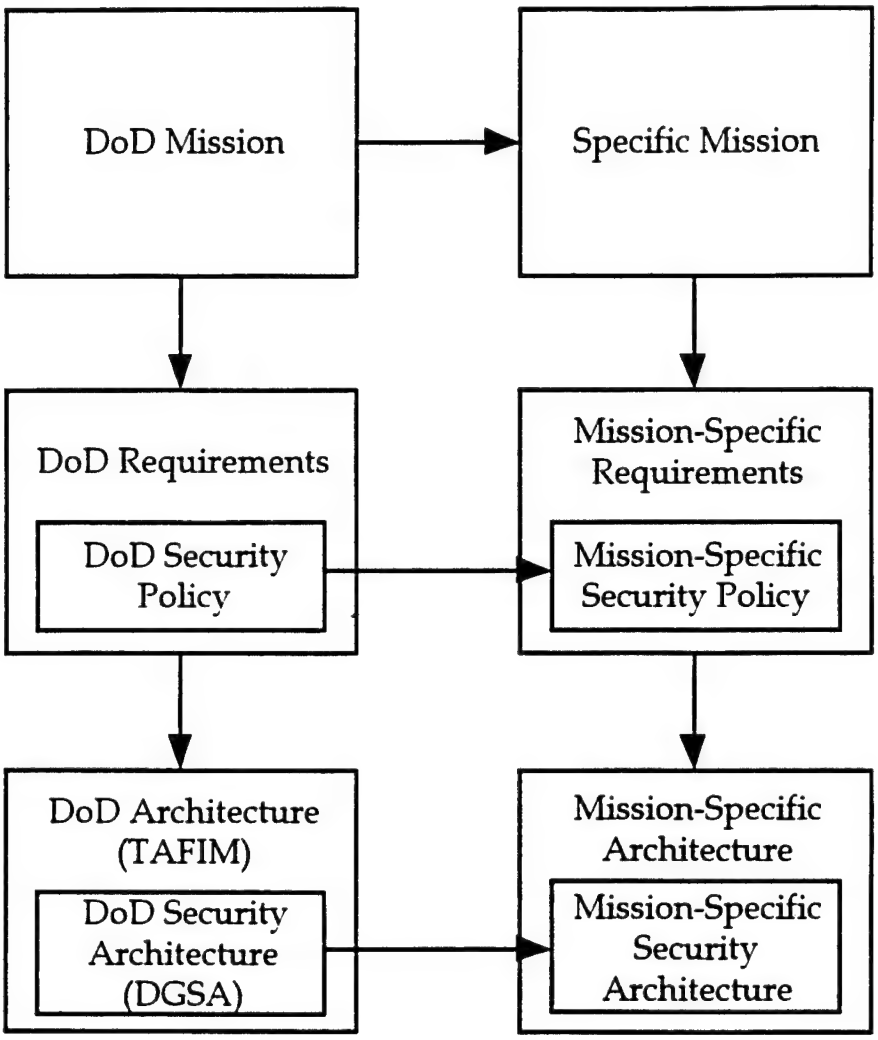

Figure 2-2. Mission-Specific Security Architecture Development

an information system needs to satisfy those requirements. The mission-specific information system security architecture is developed using this set of mission requirements and identified security services. The mission-specific architecture is stated as the set of mechanisms appropriate for providing the required protection.

Guidance documents such as the TAFIM, and particularly the DGSA, should be applied to a specific information system architecture to ensure that the necessary security protections are appropriately allocated to specific information system components. Specific security architectures also need to address any applicable policy, public laws, and executive orders. Information system security architects should understand the complete methodology and the way other aspects of the DGSA are taken into account, as demonstrated in the example in Section 8.

2.3 DOD SECURITY POLICY AND SECURITY REQUIREMENTS

1. The DISSP was initiated by appointed panels that studied various aspects of DoD information system security. Their findings and recommendations, including information processing requirements, were collected in the DISSP Action Plan. One of the recommendations resulted in the creation of a draft *DoD Information Systems Security Policy* (NSA, 1993), which is summarized as follows: DoD information systems must support information processing under multiple security policies of any complexity or type, including those for sensitive unclassified information and multiple categories of classified information.

2. DoD information systems must be sufficiently protected to allow distributed information processing (including distributed information system management) among multiple hosts on multiple networks in accordance with open systems architectures.

3. DoD information systems must support information processing among users with different security attributes employing resources with varying degrees of security protection, including users of nonsecure resources if a particular mission so dictates.

4. DoD information systems must be sufficiently protected to allow connectivity via common carrier (public) communications systems.

Notwithstanding the DISSP panels' emphasis on DoD mission requirements, reflection on the activities of other governmental and commercial organizations reveals that these policy statements also are generally applicable to them. Thus, the DGSA is widely applicable outside the DoD.

Analysis of the security policy statements above leads to a set of DGSA security requirements, including multiple information security policy support, open system employment, appropriate security protection, and common security management. These security requirements are presented at a moderate level of abstraction. There is no intention to identify every possible low-level or specific security requirement. It is expected that developers will perform similar, but complete, analyses for specific systems.

2.3.1 Multiple Information Security Policy Support

Some current information systems support simultaneous processing of information at multiple sensitivity levels (e.g., by using multilevel ssecure systems) and others support simultaneous processing of collections of information under the same security policy (e.g., Controlled Mode Workstations). However, no current information systems satisfy the long-held desire by users to operate simultaneously under several different security policies on a single device (e.g., workstation, outboard protocol device). Policy statement 1, above, recognizes that support for multiple security policy operation must become commonplace. The successful implementation of policy statements 1, 3, and 4 largely depends on the ability of information systems to separate information and user activities subject to different security policies. That is, implementations

must provide users with confidence that there will be no security policy violations when information systems are shared and the users operate under different security policies.

Security policy enforcement is dependent on the ability of supporting information systems to maintain reliably the identities of users and the identification of information under each security policy. The traditional expression of policy enforcement is that all references by users (or processes representing them) to information must be mediated by a reference monitor. The DGSA adopts and extends the reference monitor concept. (Note that any number of reference monitor implementations may be possible.)

When information processing operations take place in distributed information processing systems, the security policy enforcement for information in transit is commonly supported by mutual authentication, access control, data integrity, data confidentiality, and non-repudiation communications security services. For local (e.g., within a workstation) information processing, a similar set of security services can be applied.

2.3.2 Open Systems Employment

Employment of open systems is a typical operational requirement in many environments. Open system employment is central to providing information security among distributed DoD information systems where simultaneous support of multiple security policies is required. This requirement will lead to increased sharing of processing resources through the operation of a wider variety of applications than seen on current systems. Not only is this requirement directly derived from policy statement 2, but it supports policy statements 3 and 4 as well.

When a user seeks to perform functions in a distributed environment, the user must be able to convey information to another user (or a process) that will become the basis for decisions about what kinds of interaction will be allowed. The DGSA presumes that DoD-approved standard protocols (international or at least national or DoD standards, as opposed to industry proprietary schemes), information, and mechanisms will enable users to determine the capabilities and environment of other users or system processes with which they attempt to communicate. The determination may be made on the basis of information available before any communication is attempted (e.g., from a directory service), or as part of the initial communications service negotiation, or a combination of these approaches. The result of such a determination might be that the only common capability, within the information security policies shared by the users, is to share only non-sensitive information or that no further communication is possible.

Beyond the normal means to begin distributed processing, standards for the representation and exchange of security information are needed. Some of this information is made available as part of the communications exchanges and some is provided through security management-related exchanges. Taken together, this information is used in the provision of various security services.

2.3.3 Appropriate Security Protection

Policy statements 2, 3, and 4 refer to information systems being "sufficiently protected" or supporting users by employing varying degrees of security protection. To protect specific

information, an appropriate combination of automated, procedural, and physical methods should be chosen from the complete set employed for a particular information system within a particular environment. The appropriate security protection can only be determined by those persons responsible for the particular information and who are able to assess its value and the threats to it as expressed in the applicable security policy. The corresponding generic DGSA requirement is that specific means must be available to users to invoke security mechanisms appropriate to the task at hand.

What constitutes appropriate security protection, in part, is affected by the security protection provided by the communications system that is used among distributed systems. Policy statement 4 requires that when common carrier communications must be used, the information systems must be prepared to provide all of the appropriate security protection. The only security service that should be assumed from a common carrier communications system is availability.

The requirement for appropriate information systems security protection dictates that security mechanisms must be identified that implement security services at the level of protection required in security policies. Since some security mechanisms may be used to provide (parts of) multiple security services and some security services may be implemented by multiple mechanisms, a determination must be made that the mechanisms are appropriate individually and in combination. Initially, this is a technical activity, but the final determination involves deciding whether shortfalls in the collected security mechanisms can be accepted or whether additional measures must be put in place. This determination must be made by the users of mission information, or as is most common, the designated authority for system operation (accreditor) who represents the users.

2.3.4 Common Security Management

Like the open systems employment requirement, security management appears to be concerned with operational issues, but it actually provides the foundation for many of the security mechanisms that implement the security services chosen to satisfy the other security requirements. Commonality in security management will allow security administrators to control, in a uniform manner, systems that operate under multiple security policies in accordance with policy statements 1 and 2.

The basic elements that must be managed are users, security policies, information, information processing systems that support one or more security policies, and the security functions that support the security mechanisms (automated, physical, personnel, or procedural) used to implement security services. For each of these elements, the managed objects that constitute them must be identified and maintained. For example, users must be known and registered, the security policies must be represented and maintained, and information objects must be identified and maintained. The format for presenting the information in managed objects and operations on them must be standardized. Section 5 presents a detailed discussion of these managed objects and an architecture for security management.

2.4 FACTORS THAT CREATE ADDITIONAL SECURITY REQUIREMENTS

Several factors either directly or indirectly create additional security requirements. This section identifies selected factors that influence security and discusses the security requirements derived from those factors. The selected factors and the derived security requirements from those factors are shown in Figure 2-3. The presentation of the factors is designed to promote a thought or investigative process that should be applied to specific missions.

Operations today must exist in an environment in which major trends tend to be at odds with one another. Technology advancement has provided an opportunity to create an operational vision barely imaginable a few years ago. However, the high cost of transitions and diminishing budgets act against employing the new technologies. Intelligent strategies which may not reduce up-front costs but show valuable long-term benefits and reductions in costs will win favor. These strategies must support the long-term operational objectives of enterprises. Such strategies include portability of applications and other software, continuous upgrades of hardware and software, ensuring scalability of applications and communications resources, reuse of software components, and reuse of certification and accreditation results. Each strategy has the post-transition value of providing low-cost growth paths, if supported properly, and each strategy has an effect on security. Ease of recertification of systems and products after change may be the most important of the strategies in its long-term payoff.

2.4.1 Use of Off-the-Shelf Equipment

Economics have always been a driver in decisions to employ security solutions for information systems. Implementation of automated security measures has raised system costs while providing questionable returns on investments. One of the reasons that costs of security measures have remained high compared to their value is that security measures have been implemented in specialized, often retrofitted, components. Particularly in the face of current budgetary constraints, it is highly desirable that security features become standard elements of commercial-off-the-shelf (COTS) or government-off-the-shelf (GOTS) equipment so that security has minimal impact on price. For this change to occur, vendors must be persuaded to create products with security features that are integral parts of those products. Vendors will need to be convinced that a broad market for such products exists. Evaluation, and certification and accreditation (C&A) must become streamlined and conclusive processes so that vendors can be assured of reasonable returns on investments. Creation of a viable security product market will depend on the use of standards for commercial, international, and DoD use. Availability of COTS and GOTS products with integral security features will affect the ability to satisfy mission security requirements.

SELECTED FACTORS	DERIVED SECURITY REQUIREMENTS
Economics	Security features are standard elements of COTS and GOTS equipment
	Security product standards for commercial, international, and DoD use
	Evaluation and certification and accreditation (C&A) are streamlined and conclusive processes
Information Centralization, Access and Interoperability	Coexistence of varying sensitivities of information on the same information system
	Proper separation, authentication, labeling, and access control
Total Access to All Necessary Information	Improved authentication
	Improved availability
	General secure display implementations
Information Separation While Systems and Information are Shared Among Enclaves	Mechanisms that allow shared systems and information among enclaves, while ensuring appropriate separation of users and information
Increased Connectivity Without Increased Cost	Security mechanisms adequate to protect information from hostile entities on a network
	Standards for security protocols, authentication information, key management and distribution, security management information, voice communications, and methods to evaluate protection
Increased Access to Information and Resources	Interoperability of communications and security services
	Establishment and separation of enclaves
	Interpretation and exchange of security information in standard forms
	Management of security information
Transparency in Distributed Processing	Unitary logon and authentication
Consistent and Uniform C&A Applicable Across DoD Systems and Products	Uniform C&A procedures
	C&A results usable by evaluators and accreditors
	Metrics for effectiveness of security mechanisms
	Metrics for the interaction of a collection of security mechanisms

Figure 2-3. Selected Factors and Security Requirements

2.4.2 Objectives of Enterprise Initiatives

DoD-wide enterprise initiatives, such as the Center for Information Management (CIM) and Command, Control, Communications, Computers, and Intelligence (C4I) for the Warrior (C4IFTW), impose operational objectives that impact security. The CIM promotes information centralization, information access, and interoperability. All three of these operational objectives eliminate consideration of isolated or stand-alone implementations as a means of providing security. The derived security requirements from these objectives are the need to consider both the coexistence of varying sensitivities of information on the same information system and the provision of proper separation, authentication, labeling, and access control. C4IFTW is designed to provide the war-fighting soldier with access to any information needed to do the job, regardless of sensitivity, media, or branch of Service. Such operational objectives provide security challenges and considerations. System interfaces for war-fighting equipment are not equivalent to those for non-war-fighting equipment; thus new authentication issues are raised. Access to the information in a pull-from (information-on-demand) mode emphasizes both interoperability and availability requirements. The integration of voice, imagery, and data requires data correlation and a general secure display (windows) implementation.

While not all of the operational objectives discussed here necessarily pertain to every mission, the implications of the CIM, C4IFTW, and other relevant enterprise initiatives should be considered for their effects on specific missions.

The requirements of specific missions will, in turn, also impose requirements due to specific mission objectives. For example, most missions will require the creation of several groups or enclaves joined together to achieve some specific purpose. It is also likely that the individuals involved will be members of more than one of these enclaves and will need to operate in two or more enclaves simultaneously. Organizations can no longer afford to build separate systems to support each enclave, nor is it effective to require the user to change interface components (such as a workstation) every time the need arises to operate in a different enclave. To achieve the objective of supporting these missions, systems must ensure the separation of information while providing system and information sharing among enclaves. The derived security requirement from this mission-specific objective is to establish criteria for mechanisms that allow multiple enclaves to share systems and information while guaranteeing the separation of information and users as necessary.

2.4.3 Increased Connectivity and Access to Information and Resources

A common and significant operational objective is to take advantage of computer and communications technology to accomplish the mission at hand. This objective can be partially achieved by increasing the potential for connectivity, making additional resources available. Other operational objectives demand that such increased connectivity cannot increase cost significantly. One approach to increased connectivity is to employ commercially available, common carrier networks. However, this approach introduces significant potential risks. There is always the possibility that a hostile entity, with access to the network, will use any means affordable to mount attacks on information systems using the network. A derived security

requirement of the operational objective then, is that the security mechanisms chosen to protect information must be adequate to deter such a hostile entity.

Increased connectivity and use of common carrier systems present a perfect environment for DoD-wide interoperability. The connectivity to common carriers will dictate lower layer standard protocols (International Organization for Standardization (ISO) Open Systems Interconnection (OSI) Reference Model (RM), ISO 7498-1 (ISO, 1994a) Layer 3 and below), while the DoD missions will have to address upper layer standards (ISO Layer 4 and above) for interoperability between local environments. This standardization will include authentication information, security protocols, key management and distribution, and security management information. Similar standards for voice communications will also be needed. Additionally, the potential threat of a hostile entity will require standard methods of evaluating the protections afforded to information and other resources to ensure that remote user environments are providing equivalent protection.

As noted in Section 2.4.1, security considerations cause enclaves to arise based on mission criteria that require separation of users and information. Operational objectives, on the other hand, create the need to traverse enclave boundaries. That is, they create a need to provide users with access to any information and resources needed to complete a task. The objective includes operational concepts such as information pull, distributed processing, and information sharing. For example, pull-from may mean information will come from another enclave. Some missions will require support by non-DoD personnel and resources. This requires interoperability of communications and security services. In dealing with access to and the sharing of information and resources, the following derived security requirements must be addressed: establishment and separation of enclaves, interpretation and exchange of security information in standard forms, and management of the security information.

Transparency in distributed processing (i.e., users behaving as if all resources are locally available) is another often stated objective. Users wish to be able to be authenticated once to the local system and then transparently interact with the other systems to access resources. The derived security requirement from this objective is that information systems must have adequate local authentication schemes and security management mechanisms that free the user from the burdens of procedures such as multiple logons.

2.4.4 Achieving Uniform Accreditation

Certification is the process of determining the effectiveness of all security mechanisms. *Accreditation* is the process by which an organization (or an individual on behalf of the organization) accepts or rejects operational responsibility for an information system's performance, including security, in supporting their operations.

Certification and accreditation are complementary procedures that need to be consistent, uniform, and applicable across DoD systems and products. Certification procedures have lacked uniformity and a clear path to completion. In many cases, accreditation procedures are subjective and ad hoc. These deficiencies have caused tremendous frustration on the part of both

users and developers of systems. The results of the C&A procedures applied to particular products and systems should be immediately usable by evaluators and accreditors of products and systems that have common elements. The challenge is to develop a set of uniform procedures that establish time limits on the procedure, reduce the time to achieve product and system acceptance, and that will eliminate disparities in the C&A processes. Uniform procedures will ensure consistent and interoperable security support for an organization throughout a distributed environment.

The DGSA concepts presented in Section 3.3 provide a basis for achieving uniform accreditation. Structures and tools for information management are defined that lead to a better understanding of how information is protected, thus making C&A a more tractable endeavor.

This page intentionally left blank.

3.0 SECURITY VIEWS AND CONCEPTS

This section describes abstract and generic security views of information system architectures (Section 3.1). Security service allocations are made to the architectural components identified in these security views (Section 3.2). To accomplish these security service allocations, several concepts are presented that support the DGSA (Section 3.3). These security concepts are used throughout the remainder of the DGSA.

3.1 INFORMATION SYSTEM ARCHITECTURE SECURITY VIEWS

A typical abstract architectural view of the DIS (and many other distributed information systems) divides the information system resources into user elements and network elements (e.g., local area, wide area). This division is a useful starting point for establishing architectural views to which security services can be allocated.

3.1.1 Abstract Information System Architecture Security View

For security purposes, the most useful abstract view of the DIS groups information system resources into *local subscriber environments* (LSEs) that are connected to one another by *communications networks* (CNs). Figure 3-1 illustrates this first security view of distributed information systems.

The LSEs include all devices and communications systems under user (organization) control. The CN provides communications capabilities that allow LSEs to share information. This abstract view is useful for making certain basic security service allocation decisions, but slightly more architectural detail is necessary to make further such allocations.

3.1.2 Generic Information System Architecture Security View

The generic information system architecture security view first refines the abstract LSE and CN into several elements and then defines four generic security architecture components based on the LSE elements and the CN. These architectural components become the focus of the succeeding four sections of the DGSA (Sections 4-7).

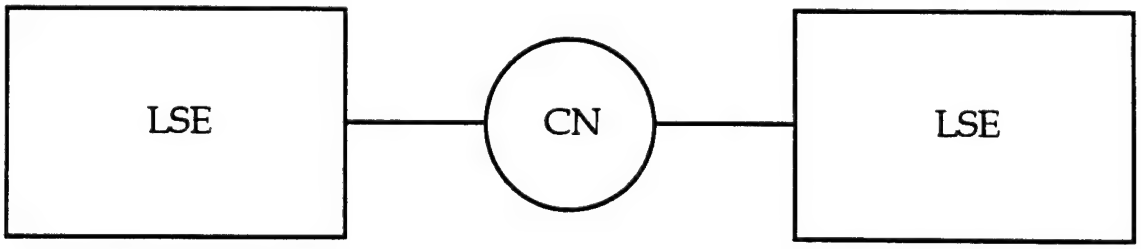

Figure 3-1. Abstract Information System Architecture Security View

3.1.2.1 LSE and CN Descriptions

Included in the LSE are three generic functional elements:

- End systems (ESs) (e.g., workstations, servers, telephones, radios, mainframes)
- Relay systems (RSs) (e.g., multiplexers, routers, switches, cellular nodes, message transfer agents)
- Local communications systems (LCSs) (e.g., rings, buses, wire lines).

The principal distinction between end systems and relay systems (as described in ISO 7498-1) is that end systems support users with direct human interfaces and personal or general applications, while relay systems are only indirectly accessible by users and the functionality is limited to information transfer relay functions. Some relay system functions may be performed in many communications protocol layers (see Section 6). LCSs serve to connect ESs and RSs within an LSE. LCSs may consist of a variety of components, but generally the DGSA is not concerned with specific technologies. Where necessary, the abstract CN can be refined to generic elements such as packet switches, routers, and transmission elements. Generally, the DGSA is independent of particular switching element and transmission technologies, so it is usually adequate to refer to a CN as both an abstract and a generic element.

An LSE may contain a single end system such as a workstation, a single relay system such as a router, or combinations of end systems and relay systems connected through LCSs. All physical elements of the information system architecture are either part of an LSE or are CNs. This security view does not imply that LSEs are only connected to one CN or that they are connected only in pairs.

3.1.2.2 Generic Security Architecture Components

From a security perspective, it is not enough to consider only the physical information system elements. It is necessary to take into account the environment in which the elements are employed and the means through which they are managed. The resulting generic security architecture view includes four components to which security service allocations will be made:

- ESs and RSs - information processing elements
- Security management - security-related activities of information system management
- Transfer system - LCS and CN elements and communications protocols used by them and by ESs and RSs
- Physical and administrative environment - security related to environmental (physical) elements and personnel.

Figure 3-2 illustrates a generic view of several LSEs joined by CNs. Each LSE is defined and bounded by the elements under user (organization) control, including the environment. LSEs exhibit all or parts of each of the four generic architecture components, while the CN only represents a part of the transfer system.

End systems and relay systems are entirely contained within an LSE. Although Figure 3-2 shows ESs and RSs as separate generic components, in practice the same information system may combine both ES and RS functions as necessary. LSE connections to CNs are only through RS functions.

The security management component is not illustrated in this figure, but its functions are pervasive in the LSEs and extend to cooperate with CN management facilities.

The transfer system component is shown within dashed lines. Although it includes all of the LCS and CN elements, it includes only these portions of the ESs and RSs that implement communications protocols.

The physical and administrative environment component (labeled collectively as *environment* in Figure 3-2), represents all of the generic security services provided directly or indirectly by physical means (e.g., locked gates, guard dogs) or through administrative procedures (e.g., background investigations, issuance of badges).

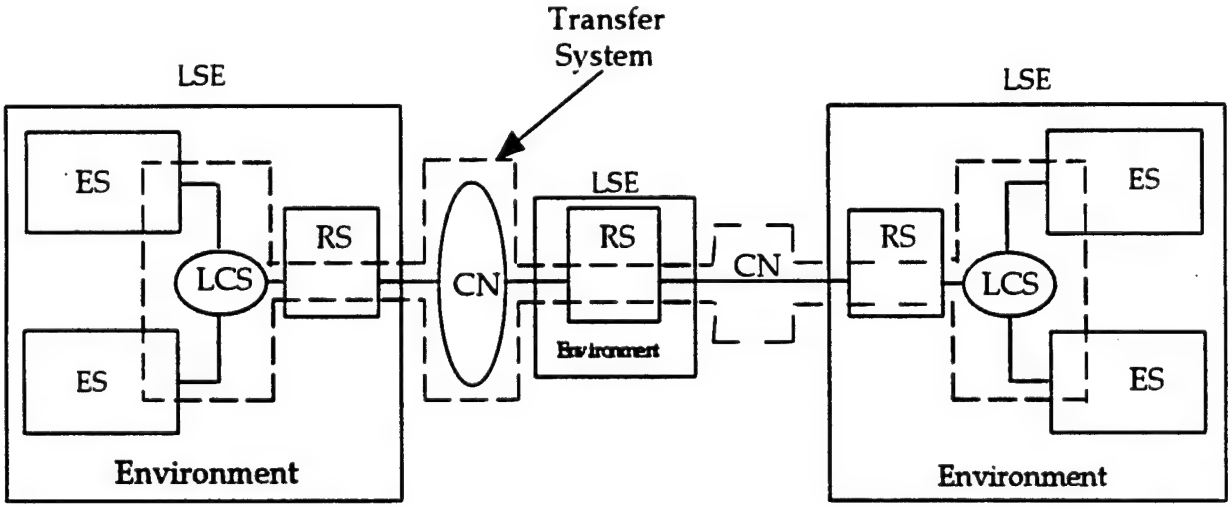

Figure 3-2. Generic Security Architecture View

3.2 SECURITY SERVICE ALLOCATIONS

The DGSA's security services are based on those defined in ISO 7498-2 (ISO, 1989a) for data communications. These security services include authentication, access control, data integrity, data confidentiality, and non-repudiation. (The *OSI Security Frameworks*, ISO 10181, is a multi-part standard that discusses each of these services, plus security audit and key management, in considerable detail.) In the DGSA, availability also is considered to be a basic security service. The basic security services are considered to apply not only to the transfer system, but are interpreted to apply to the entire LSE. This section discusses security service allocations to CNs and LSEs, and to the four generic security architecture components.

3.2.1 Abstract Architecture Security Service Allocations

In this section, security service allocations are made to the abstract security architecture components.

3.2.1.1 CN Security Service Allocation

In response to the requirements of Section 2, particularly the requirement to use common carrier services, the DGSA makes only a security service allocation of communications availability to CNs. CNs must provide an agreed level of responsiveness, continuity of service, and resistance to accidental and intentional threats to the communications service.

The reliability, flexibility, contingency actions, management, and preventive maintenance of CNs are some of the factors that will determine the availability of communications services. Protection of CN resources from accidental or intentional damage is both a security concern and, in the commercial world, a direct financial concern. Well-designed and well-managed CNs should exhibit graceful degradation in service and should provide for establishing priorities of service. CN providers will employ various security services to protect the CN's own resources to ensure that the agreed availability will be maintained. However, CNs are not relied upon for the confidentiality or integrity of the information they transfer. Failures in CNs can only result in the delay, misdelivery, or non-delivery of otherwise adequately protected information. The purpose of CN management, which is to counter these failures, is identical to that of the security service of availability.

3.2.1.2 LSE Security Service Allocations

All the security services are allocated to LSEs. The provision of security services for an entire LSE is accomplished by physical, administrative, and personnel security mechanisms. Physical LSE boundaries can limit facility access to authorized personnel. Protection of LSEs is provided in part by the logistical support system (e.g., configuration management control). In turn, LSEs provide protected environments for their end system, relay system, and LCS components. (See Section 7 for additional details.)

The open systems requirement of Section 2 demands that LSEs with highly sensitive information must have the ability to communicate with nonsecure as well as with secure LSEs. The architectural model for such LSEs is shown in Figures 3-3 and 3-4.

In Figure 3-3, a secure LSE is communicating with a nonsecure LSE that must be assumed to include hostile entities if the total information system is truly open. In this situation, no transfer system security services are used to protect information in transfer because none are needed and the nonsecure LSE offers no such services. The secure LSE must isolate its sensitive information (shown as shaded in the figure) and protect it with its own security mechanisms.

In Figure 3-4, both LSEs are considered secure (for at least some set of information) and cooperate to provide transfer system security services to protect the information in transfer. The secure LSEs must still protect themselves from nonsecure LSEs that are connected to the CN. The requirement for open systems provides serious challenges to the security architecture of LSEs.

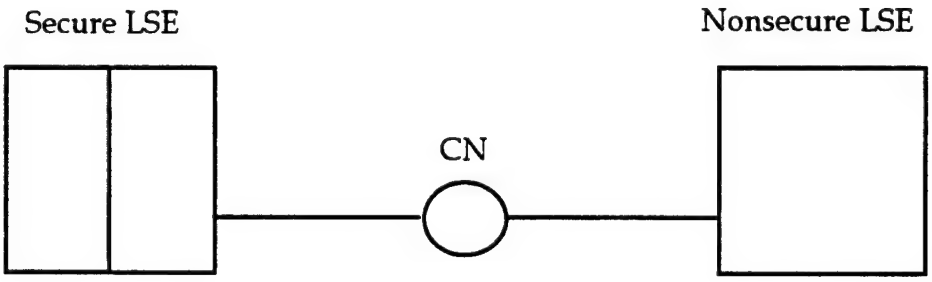

Figure 3-3. Secure-to-Nonsecure LSE Communications

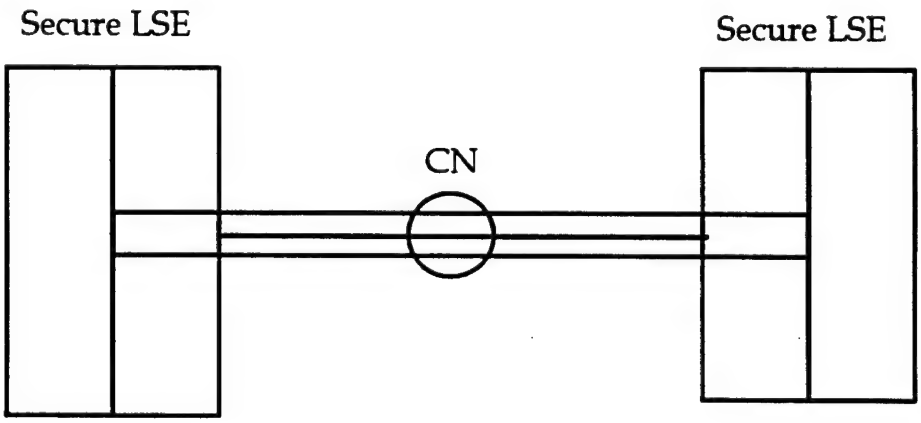

Figure 3-4. Secure LSE Communications

3.2.2 Generic Architecture Security Service Allocations

In this section, security service allocations are made to the generic security architecture components.

3.2.2.1 End System and Relay System Security Service Allocations

Security service allocations are made to end system and relay system hardware and software so that the hardware protects the software and the software protects information being processed, transferred, or stored. End system and relay system hardware and software collectively provide the security services of user identification and authentication, access control, data integrity, data confidentiality, non-repudiation, and availability. Details of the end system and relay system security architecture are discussed in Section 4.

3.2.2.2 Security Management Security Service Allocations

All the security services are allocated to the security management component, but only indirectly. The function of the security management component is to support and control the other architectural components. Security management applications and protocols are simply a portion of end system and relay system hardware and software compositions. Section 5 presents details of the security management architecture.

3.2.2.3 Transfer System Security Service Allocations

CNs have already been allocated the availability security service. LCSs are required only to provide the availability security service for communications among end systems and relay systems within LSEs. Other security services may be provided in LCSs for local purposes if they do not interfere with other requirements, such as interoperability with other LSEs.

Security services implemented within protected end systems and relay systems provide the basis for the protection of information being transferred. The remaining security service allocations to the transfer system make it responsible for peer entity and data origin authentication, access control, non-repudiation, confidentiality, integrity, and availability of information in transfer. The protection of information being transferred enables the protected distribution of security-relevant information for security management as well as user information. The sharing of identification and authentication information, audit records, key management information, and policy and privilege management information among LSEs can be safely accomplished if the transfer system is protected. Section 6 provides additional detail on the transfer system architecture.

There is a particular aspect of data confidentiality, usually referred to as *traffic flow security* (TFS), which is the responsibility of the transfer system. True TFS only can be provided by a class of security mechanisms that inherently conflict with some of the security policy statements (Section 2.3) upon which the DGSA is based. Under certain circumstances, it may be judged that the threats to a mission can only be countered using TFS. Because TFS mechanisms are costly and because some goals (e.g., interoperability) will be sacrificed to some degree, the

employment of the TFS service must be carefully considered. See Section 6.3.1 for additional discussion of this topic.

3.2.2.4 Physical and Administrative Environment Security Service Allocations

All security services are allocated to the physical and administrative environment architecture component. Specific mechanisms to implement these services that protect the LSE are discussed in Section 7.

3.3 SECURITY CONCEPTS

The most significant capabilities of the DIS target architecture are distributed processing and open communications. The objectives for security in such an environment are to maintain open and distributed capabilities and yet be able to establish and enforce a wide range of mission and information security policies. A simple characterization of such an environment is that resources and information may be shared or isolated as desired.

The management of information is accomplished by individuals and groups of people who create, collect, process, categorize, store, transfer, and communicate particular information. The value of that information and, therefore, the required protection of that information is determined by the group. The group determines the conditions for authorized access to the information and the conditions for individuals to become members of the group. This approach applies equally to United States national classified information, trade secrets, proprietary data, or other identified collections of government, corporate or personal information. Three elements are necessary for this idea to be employed:

- A group must have a defined membership
- Information objects must be uniquely identified within the domain of the group
- The security policy regarding the protection of and access to the information objects must be known and agreed to by the membership.

Several concepts have been developed to support this approach to information management. They are information domains, strict isolation, and absolute protection. The ways in which these concepts influence and are supported by the DGSA generic architectural components (end systems and relay systems, security management, transfer system, and physical and administrative environment) are detailed in Sections 4, 5, 6, and 7.

3.3.1 Information Domains

An *information domain* is a set of users, their information objects, and a security policy. An information domain security policy is the statement of the criteria for membership in an information domain and the required protection of the information objects. Information domains

are not hierarchically related, nor may they implicitly or explicitly infer a sensitivity relative to multiple categories of sensitivity.

In contrast to domains that might be composed of systems or networks, information domains are not bounded by systems or even networks of systems. Information domains are bounded by the presence of their identifiable information objects and may be supported by any information system that can meet the protection requirements of the information domain security policy. In this concept, a specific mission security policy may define several information domains, each with its own distinct information domain security policy. The security mechanisms of any number of information systems may be evaluated for their ability to meet these information domain security policies. Through the process of accreditation, these security mechanisms may be usable for part or all of one or more missions.

Each information domain is identified uniquely. The unique identification indicates (directly or indirectly) the sensitivity of all the information objects in an information domain. Any security-relevant attributes and attribute values of information objects in an information domain must be the same for all information objects in the information domain. That is, there must be no security-relevant distinction made among the information objects in an information domain. Members of an information domain may have different security-related attributes and attribute values. For example, some members might have only read permission for information objects in an information domain, while other members might have read and write permissions. Since all information objects in an information domain have the same security-relevant attributes and attribute values, a user who has read and write permissions in an information domain has those permissions for *every* information object in the information domain.

3.3.1.1 Interdomain Information Sharing and Transfer

Some mission requirements will necessitate the sharing or transfer of information objects among information domains. The establishment of new mission functions, new mission area relationships, or new organizations are examples of events that can create requirements for information sharing and transfer.

The simplest method of sharing information is to accept new members into an existing information domain and to grant access privileges to them. Where a need exists to share some, but not all, of the information objects in one or more information domains with members of other information domains, a new information domain may be created to contain the shared information objects. The new information domain, like any other information domain, requires a security policy. The members of the new information domain may or may not be members of the information domains from which its information objects were obtained.

Information objects can be transferred between two information domains only in accordance with established rules, conditions, and procedures expressed in the security policy of each of them. The transfer can be accomplished only by a user who is a member of both the sending and receiving information domains and, if required by the information domain policies, has been granted the appropriate privileges (e.g., "release authority").

The transfer of information objects between information domains may be implemented as a move operation (in which the information object no longer exists in the originating information domain), or as a copy operation (in which the information object exists in both information domains). Information objects moved or copied from one information domain to another must be relabeled with the label of the information domain to which the information object has been moved or copied.

In general, interdomain transfers can only occur within an end system or relay system. Interdomain transfers usually cannot occur among distributed end systems or relay systems; transfers among end systems or relay systems usually can only occur within the same information domain. These restrictions are consequences of the nature of security contexts and security associations that are used to create an appropriate environment for distributed information domain operations (see Section 6.1.3.2).

3.3.1.2 Security Contexts

A *security context* encompasses all end system resources and security mechanisms (including physical and administrative) that support the activity of a user operating in an information domain. When the end system ceases performing operations in one security context and begins performing operations in another, information cannot be allowed to pass from one security context to another unless a specific request is made. Also, communications among security contexts in an end system can only take place in accordance with the security policies of the information domains supported by the security contexts. Each information domain security policy must include a transfer policy which defines under what conditions information may move from one security context to another.

3.3.1.3 Security Associations

To support distributed processing, it is necessary to establish security contexts for the same information domain in the cooperating end systems. These contexts must communicate with one another with the same assurance as if they were in the same end system. A *security association* is the totality of communications and security mechanisms and functions (e.g., communications protocols, security protocols, security mechanisms and functions) that securely binds together two security contexts in different end systems or relay systems supporting the same information domain. A security association extends the protections required by an information domain security policy within an end system to information in transfer between two end systems. It also maintains strict isolation (see Section 3.3.2) from other information domains.

3.3.1.4 Multidomain Information Objects and Policies

The missions of most organizations require that their members operate in more than one information domain. The information management activities of a mission may be viewed as taking place in a set of information domains, some of which may be shared with other missions. To carry out their mission information management activities, users may need to process information objects from several information domains concurrently. Often, a user may have a

perception that a collection of information objects from different information domains is a single, composite information object. Such a composite information object is referred to as a *multidomain information object.* This perception must be achieved without actually combining real information objects from different information domains to create real multidomain information objects. When creating the perception of multidomain information objects, strict isolation among information domains must be maintained, and the constituent information objects within the multidomain information object must be managed only in accordance with their individual information domain security policies. The purpose of multidomain information objects is to be able to define a collection of information objects to be displayed, printed, or transferred between information systems in a particular order or arrangement.

The creation and use of multidomain information objects must be subject to some security policy. The simplest policy is the one noted above, namely to conform to the policies of the individual information domains. However, in such cases it may not always be possible to print such a multidomain information object or to convey it to another user or information system. A multidomain information object security policy might be based upon some existing policy (e.g., U.S. national security policy) that states a relationship among the constituent information objects. Such a security policy for multidomain information objects is made part of the security policy of the information domains of the constituent information objects. In situations where the security policy for multidomain information objects is complex or involves several information domains, that security policy might be stated in one place in the supporting information system and be referred to by the individual information domain security policies.

Explicit multidomain information object security policies must state the specific privileges a user must have to view, print, create, delete, or transfer a multidomain information object between information systems. To create or otherwise deal with an entire multidomain information object, the user must be a member of each of the information domains in which the constituent parts of the multidomain information object are located. Some multidomain information object security policies might allow access only to the component parts of a multidomain information object for which the user has appropriate privileges, but in many cases this would not result in a sensible multidomain information object.

As noted in Section 3.3.1.1, information domains are not hierarchically related. Nonetheless, security policies for multidomain information objects may recognize marking rules that apply to the entire multidomain object or its parts based on existing policies (such as paragraph and page markings for information subject to U.S. national classification policy) when printed or displayed. Further, an information domain security policy is not precluded from recognizing that a user security clearance of Top Secret is adequate for access to the information objects in an information domain that contains U.S. national classification Secret information objects, if all other aspects of the information domain security policy are also met. (Note that the apparent hierarchy among U.S. national security policy classifications is actually a property of user privileges, in the form of clearances, rather than a relationship imposed on information of different classifications. Information that is classified Secret is *not* a subset of information that is classified Top Secret.)

The implementation of multidomain information objects in real information systems has many implications for end system, security management, and transfer system architectures. These implications are discussed further in Sections 5, 6, and 7.

3.3.2 Strict Isolation

The diversity of missions and the threats to the security of their information will result in information domain security policies with unrelated protection requirements. Thus, information systems that support multiple information domain security policies must adopt a protection strategy that provides a basis for satisfying all of them. One such strategy, termed *strict isolation*, is to isolate one information domain from another, except when there is an explicit relationship established. Under this strategy, an information system must provide mechanisms that maintain separation of information domains in ways that are satisfactory to each of them. The default information system security policy is strict isolation among the information domains supported.

In the absence of any information domain security policy to the contrary, an information object must be isolated. While such a situation is a logical possibility, in practice, all information objects should belong to an information domain that has a defined membership and an information domain security policy. Information domains with no explicit interdomain policies must adopt a policy of strict isolation to be enforced by the systems that support them.

3.3.3 Absolute Protection

Since open systems may consist of an unbounded number of unknown heterogeneous LSEs and it may be necessary to communicate with any of them, system security architects must have a rational basis for protection decisions in such an environment. In this environment, it is not possible to rely upon the assurances provided by physically separated networks or cryptographically isolated LSEs. Information domains must rely on the protections afforded by a heterogeneous collection of LSEs. The concept of *absolute protection* (which does not imply perfect protection) is set forth to provide a framework for achieving uniformity of protection in all information systems supporting a particular information domain. It directs its attention to the problems created by the interconnection of LSEs that provide disparate strengths of security protection.

In order to support an information domain in multiple LSEs, the overall strength of protection afforded to information objects must be consistent in those LSEs. Strength of protection is a function of the strength and correctness of security mechanisms (including physical and administrative environment) implemented in LSEs to satisfy an information domain security policy. The required strength of protection is determined by assessing the value of the information being protected and then assuming a hostile attacker has logical access to the LSE through the transfer system. *The specific mechanisms and their implementations need not be identical in every LSE that supports an information domain, but the implementations must provide at least the required strength of protection.*

If the overall strength of protection provided by each LSE supporting an information domain is successfully evaluated under the assumption that the LSE is logically accessible to a hostile user, then each of these LSEs can be accredited as being adequate to protect the information domain against the same threats. Protection provided in all the accredited LSEs under these conditions will be absolute, non-relative, and equivalent. Absolute protection is primarily concerned with the vulnerabilities created by connections to communications networks. This concept generally forces stronger mechanisms to be employed for information of a given sensitivity.

For system security architects, implementors, and accreditors to properly apply the concept of absolute protection, different approaches to evaluation of security mechanisms, components, and information systems will be required to determine equivalent protection. A single measure of overall strength of protection is not adequate. Rather, security mechanisms will need to be rated (measured) for their ability to support one or more security services, alone and in combination with other security mechanisms. The required strength of protection for an information domain will be translated to a set of such measures so that an appropriate set of security mechanisms can be chosen. This method of choosing security mechanisms will give security architects, implementors, and accreditors a consistent means for providing equivalent (though not necessarily identical) protection in the LSEs that support an information domain.

4.0 END SYSTEMS AND RELAY SYSTEMS

A generic security architecture for end systems and relay systems must be appropriate for a wide range of applications and environments. Among the many possible implementations, some unifying structure must be created that permits a generic approach to security. This structure must accommodate the requirements of Section 2 and the primary security allocations made in Section 3. This section refines several concepts presented in earlier sections for end system and relay system architectures, including security allocations, types of functions that are required to support the security allocations, types of devices that make up end systems and relay systems, and technologies that should be considered in specific implementations. Section 4.1 gives an overview of the end system security architecture, and its description is presented in Section 4.2. Section 4.3 lists candidate technologies to support implementations. Generally, relay systems provide services that require the same kinds of underlying support as end systems, except that they do not provide support for direct user interactions. Thus, a single security architecture for end systems and relays systems is appropriate. The remainder of this section refers to both end systems and relay systems simply as *end systems*.

Since the DGSA is a generic architecture, not all of its possible architectural choices and alternatives (security services and mechanisms) will be used in every specific implementation. The DGSA allows for a wide variety of specific implementations that will be dictated by missions and threats. Similarly, the generic end system security architecture must have wide applicability. The end system security architecture described here is a current best estimate of how the DGSA requirements can be met. To the extent that it depends on specific technological directions, it is subject to change as experience and technology dictate. However, the basic architectural decisions described should remain stable.

Much of the end system security architecture is similar to that proposed by Rushby (1984). There are some significant departures from Rushby's proposal, most notably with respect to centralization of security policy-related functions. Rushby argues for such functions to be tailored to and to be implemented with specific resource management functions. This argument is implicitly based on the fact that only a single, access control-based security policy is to be enforced. The DGSA requirement for supporting differing security policies per information domain (which may have other dimensions than simply access control) makes the argument for centralizing the basic security policy-related functions more attractive. More recent proposals for support of multiple security policies suggest architectural approaches which take a middle ground and may offer some performance advantages (see Abrams (1993) for a summary and extension of these approaches).

The end system security architecture focuses on conventional computer systems, which represent a large portion of all end systems. Other end system types may need to implement only portions of the end system security architecture. In extreme cases, such as simple sensor devices, the end system functions may be so limited that only specialized implementations of a small portion of the end system security architecture are appropriate (for example, such a device almost certainly would not need to support multiple information domain security policies).

4.1 END SYSTEM SECURITY ARCHITECTURE OVERVIEW

In Section 3, fundamental allocations of security services were made to LSEs and to the end systems and LCSs within LSEs. Security service allocations were made to LSEs to protect their resources, including end systems. The end system security architecture makes additional security service allocations to the end system hardware and software. Not every security service allocation needs to be made identically in every system. For example, if electronic emanations are considered to constitute a potential vulnerability, the responsibility for countering it could be assigned to the LSE or to one or more of its components. Similarly, there is flexibility with regard to how protection responsibilities are shared between end system hardware and software.

4.1.1 The LSE Protects the Hardware

As discussed in Section 3, the security service allocations to the LSE are implemented as physical and administrative security mechanisms. Administrative and environmental security mechanisms are discussed in more detail in Section 8. The primary security service allocations to the LSE are access control to facilities and some aspects of authentication of personnel. In addition, some aspects of information confidentiality and integrity, and system integrity and availability may be allocated to the LSE.

4.1.2 The Hardware Protects the Software

Section 3 assigned responsibilities to the end system for all security services. There are a variety of security mechanism choices available between the hardware and software portions of the end system, but certain general allocations and properties can be stated for the hardware.

The hardware is relied upon to function correctly, to enforce isolation of software functions, and to contribute to the protection of the integrity of the system applications and the operating system. It provides protected paths between users and trusted parts of the software. The hardware indirectly supports the isolation of information processed and stored in the end system by protecting the integrity of the software. Hardware mechanisms are used to protect the system from radio frequency interference and to prevent undesired emanations. In some environments, specific hardware technologies (e.g., protective coatings, hardened or alarmed containers) may be necessary to protect against tampering with end system components. Availability of an end system may be enhanced through technologies such as fault-tolerant and fault-detecting hardware features. Hardware cryptographic mechanisms are employed as needed to support various security services. Other hardware mechanisms (e.g., memory mapping) support specific aspects of the software architecture and are noted in the end system security architecture discussion (Section 4.2). There is an array of equipment available to support the hardware allocations.

4.1.3 The Software Protects Information

The security service allocations made to software are wide ranging. The portion of the transfer system supported by the end system software is responsible for the confidentiality and integrity

of information transferred among end systems, for the authentication of end systems to one another, and for user authentication and access control in distributed systems. The details of how the transfer system is supported by end systems are presented in Section 6.

Security services and the mechanisms that implement them must be managed. The software applications that support security management in end systems are discussed in Section 5 and are extended in Section 6 for transfer system support.

The end system software is responsible for user authentication and access control, and for the integrity of information being processed and in storage. Correct operation of certain software is required to ensure end system availability. Additionally, the software is expected to provide functions that support the security policies and requirements stated in Section 2 that are not directly expressed as security services, such as support for multiple security policies. The remainder of this section refines the end system security architecture, which primarily is concerned with software structure.

4.2 END SYSTEM SECURITY ARCHITECTURE DESCRIPTION

A generic end system security architecture must respond to the security allocations discussed earlier, and it must be sufficiently flexible to encompass changing technology. The end system security architecture presented in Figure 4-1 is an example and not an implementation specification, and might be realized in several ways. The end system security architecture concentrates on support for multiple information domains with distinct security policies. Attention is paid to strict separation of information domains, management of end system resources, and controlled sharing and transfer of information among information domains. The end system security architecture also relies upon an engineering approach that seeks to isolate security-critical functions into relatively small modules that are related in well-defined ways. This approach has advantages in implementation, certification, and accreditation by limiting the scope of particular portions of these activities. While there are no existing end systems that specifically implement all of the end system security architecture, several efforts have been documented in the academic and research communities that support various aspects of the end system security architecture. Recently, commercial operating system vendors have adopted design and implementation strategies that share significant aspects of the end system security architecture.

A *security context* is a combination of all the LSE, hardware, system software, user application software, and information supporting the activities of a user (or system function) operating in an information domain. A security context builds on the common operating system notion of a user process space (sometimes called a *context*) as supported by hardware features and operating system functions. The primary distinctions between an ordinary user process space and a security context are that aspects of protection provided by the LSE are explicitly included, and that user applications operate in a controlled process space *subject to an information domain security policy*. Security contexts are described in more detail in Section 4.2.2.

A *separation kernel* manipulates the protection features of the end system hardware (e.g., processor state registers, memory mapping registers) to maintain strict separation among security contexts by creating separate address spaces for each of them. A separation kernel also controls communications among security contexts to allow sharing or transfer of information, and to allow services to be performed by one security context for another. All user security contexts and many system function security contexts are constrained to make requests for basic end system services on the separation kernel through a *standard kernel interface*. The separation kernel is described further in Section 4.2.1. The functions that make and enforce security policy decisions are intimately related to the separation kernel. These are described in Sections 4.2.1 and 4.2.3.1.

In Figure 4-1, end system software is divided into trusted (shown in the shaded area) and untrusted parts for practical evaluation. The trusted parts of the software are those that are considered so important to the secure operation of the end system that they must undergo strict evaluation procedures and come under strict configuration management control.

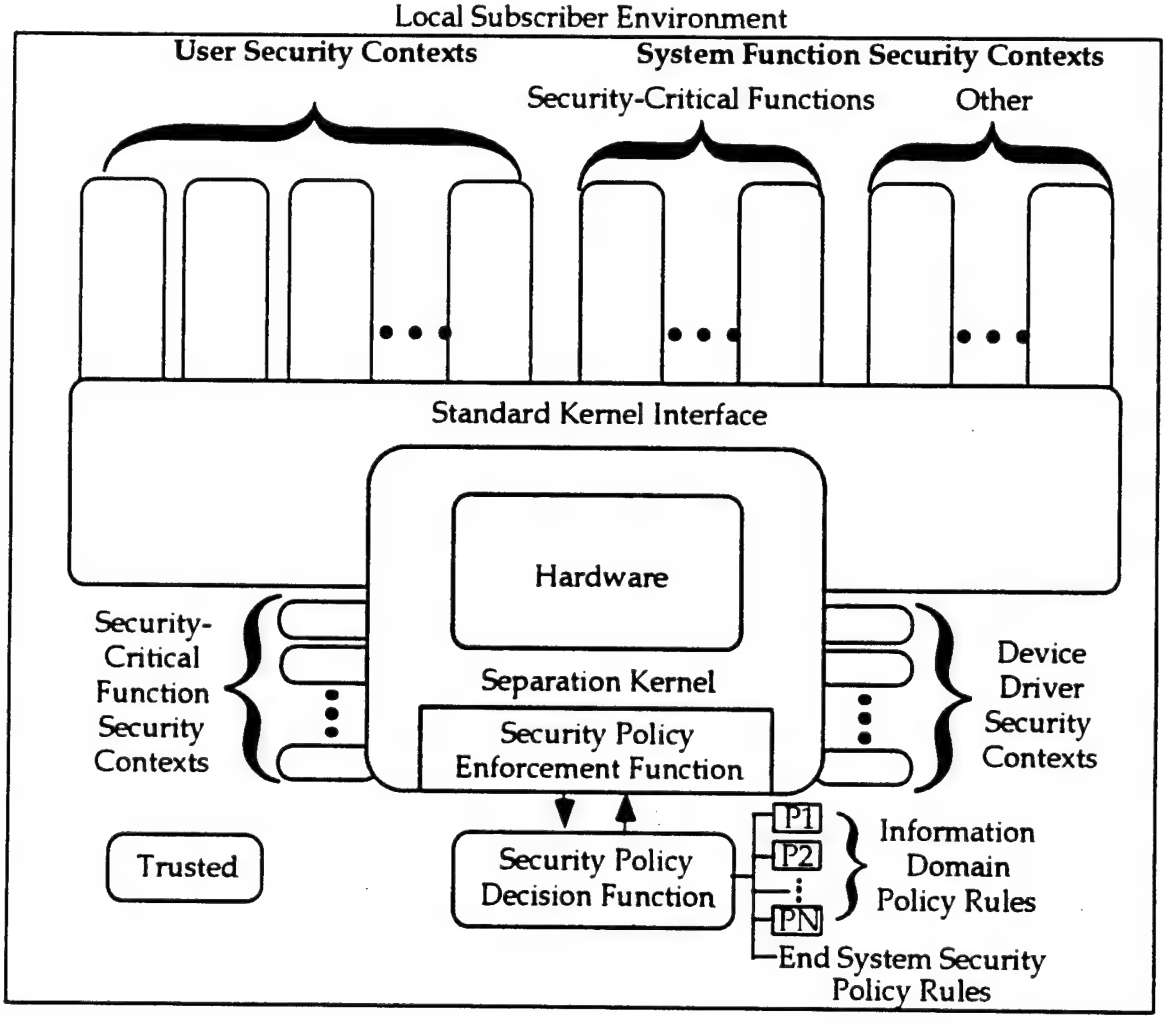

Figure 4-1. End System Security Architecture Generic View

The hardware (including any microcode) is considered trusted in the sense that its operation is assumed to be correct. Untrusted software is able to perform operations on basic system resources only through invocations of security-critical functions that are mediated by the separation kernel; inter-security context operations (e.g., inter-information domain communications) are performed by security-critical functions.

Untrusted *security-related* functions (such as security management applications and portions of transfer system applications) are expected to operate correctly to satisfy user operational needs, but need not be subjected to the rigorous scrutiny applied to the security-critical functions. Security-related software is not assumed to be free of security defects, although it is certainly prudent to obtain such software from reliable sources, test it before use, apply integrity safeguards to ensure it remains unchanged, and apply configuration management to it. (Software obtained from less than reliable sources may need to be inspected more carefully.) Under these conditions, if faulty application software is introduced into a system it will, at worst, prevent certain operations, but information compromise will not result because of the combination of strict isolation of information domains enforced by the end system, testing, and configuration management. The remaining software is not only untrusted, but is not expected to be examined for any security reasons.

The following subsections provide additional detail on the end system security software components, primarily for the separation kernel, security contexts, security-critical functions, and operating system implementations.

4.2.1 Separation Kernel

Much general operating system research has concentrated on organizing basic operating system functions into a collection called a kernel. The kernel presents abstractions of the fundamental resource management mechanisms to other, less primitive, service providers (information system functions and applications). In operating system implementations that attempt to provide a basis for secure information processing, the kernel software is carefully constructed and evaluated. To aid the evaluation process, the kernel functions are implemented as relatively small programs that are independent of one another to the maximum extent possible.

Rushby suggested that significant improvements in secure operating system kernel design and implementation could be achieved by isolating each kernel function in its own process space (i.e., address space). The benefit of this approach is that each operating system function performs a single, well-defined activity and can be understood and evaluated in relative isolation from all other functions. A separation kernel is charged with the critical task of providing separation among process spaces by manipulating the protection features of the end system hardware.

Until recently, most secure operating system designs have been limited with regard to security policy specification and enforcement. Particular limitations include support for only a single security policy (usually an access control policy) and the inability to change security policy conveniently. The end system security architecture adopts a particular view of operating system

kernel design to meet DGSA requirements and concepts, most notably the support of multiple security policies so that a single end system can support users in different information domains simultaneously. The traditional operating system kernel functions are divided among the separation kernel, security policy enforcement and decision functions, and the remainder of the trusted operating system functions, called the *security-critical* functions. The separation kernel serves as the ultimate security policy enforcement function by mediating all use of the basic information system resources. The separation kernel notion is the foundation of the end system security architecture. However, any other information system mechanism that provides equivalent isolation of information domains and control of system resources is appropriate for implementations that are consistent with DGSA objectives.

The end system security architecture generalizes an approach that is becoming widely accepted concerning access control, namely the independence between the decision of whether or not an access to a resource is allowed and the enforcement of that decision. The separation of access control decision-making and access control enforcement functions allows the support of multiple access control policies. The ISO Access Control Framework (ISO, 1995d) designates these functions the *access control decision function* (ADF) and the *access control enforcement function* (AEF), respectively. In fact, most existing secure operating system designs have concerned themselves only with access control policy. Since one of the DGSA requirements is to support any security policy, the end system architecture extends the AEF concept to include the enforcement of all aspects of an information domain security policy. The resulting function is called the *security policy enforcement function* (SPEF). Similarly, the ADF concept is extended to a *security policy decision function* (SPDF). (The SPDF is discussed in more detail in Section 4.2.3.1.) The separation kernel is the implementation of the SPEF in the end system security architecture.

The separation kernel also is an extension (beyond access control) of the *reference validation mechanism* (RVM) described in the Trusted Computer System Evaluation Criteria (Department of Defense, 1985). The basic properties of the RVM must be applied to any separation kernel implementation: it must be invoked for every security-critical operation, it must be small enough to be verified, and its integrity must be maintained.

In the spirit of several current standardization efforts, a standard kernel interface will be defined to allow open system development of operating systems and applications built on implementations of the DGSA end system security architecture. The standard interface to the separation kernel is the same whether the underlying computer is a large multiprocessor mainframe or a single-processor workstation. This approach allows developers great latitude in implementing the separation kernel and the security-critical functions.

4.2.2 Security Contexts

From the perspective of the separation kernel, a security context is defined by a set of data and programs operating in accordance with an information domain security policy. As noted earlier, a security context also includes the physical and administrative security mechanisms of the LSE, and the hardware-based resources (e.g., registers, memory, disks) that are in use when the end

system is serving a particular user (or system function). That is, a security context encompasses all end system resources and security mechanisms that support the activity of a user operating in an information domain. The separation kernel must maintain all the information needed to isolate one security context from another. When the end system ceases performing operations in one security context and begins performing operations in another security context, no information can be allowed to pass from one security context to the other unless a specific request is made and it is allowable under the security policies of the information domains involved.

Examples of information that end system security-critical functions (including the separation kernel) must maintain to support the operation and isolation of security contexts include:

- A unique identification for each security context
- The identification of the information domain being supported
- Hardware register values related to control of end system resources, including virtual memory and all devices in or attached to the end system
- The authenticated identity of the user being served
- The user's security attributes (permissions)
- Data structures needed to operate security-related functions and other untrusted system applications.

Each security context supports a user (or a system function) operating in a particular information domain. Over a period of time, an end system may maintain several security contexts to support one or more users operating in one or more information domains. A particular user might use (simultaneously or serially) security contexts operating in the same or different information domains. Different users may employ security contexts operating in the same or different information domains.

Since security contexts are isolated from one another by the separation kernel, communications among security contexts (requests for service or information transfer) in an end system can only take place in accordance with the security policies of the information domains supported by the security contexts. If the security policies of the supported information domains do not explicitly permit inter-information domain transfer, the SPDF will necessarily deny the request and the separation kernel will enforce that decision. Since an information domain contains the information of a particular user community, it would be unusual for an information domain security policy to prohibit information sharing between two security contexts supporting the same information domain.

Many end system activities are not carried out on behalf of a specific user (either an individual or the entire membership of an information domain as a group), but rather for basic end system operation and management. Examples of such activities include many of the security-critical

system functions and end system management activities. These activities are carried out within end system security contexts on behalf of one or more of the information domains supported by the end system. The security policies of these end system information domains are created to exercise appropriate control of end system resources for all of the user information domains supported by the end system. Some example uses of end system information domains include the control and manipulation of multidomain objects, login applications, and management information domains.

Multidomain information objects (see Section 3.3.1.4) never exist in an end system except as displayed (or printed). Nonetheless, in end system implementations, it must be possible for a user to describe the relationships among the components of a multidomain information object so it can be displayed. Some implementations of multidomain information objects will result in the description being represented as an information object. Some security policies may preclude this information object from being held in any of the component information domains. In such cases, the end system must be able to create a system security context in which the description can be used by an appropriate application program that requests the display manager to construct the multidomain information object on a display device. Note that the multidomain information object description could be retained by the end system for future use by either the creator of the description or by other users who have the necessary information domain memberships. Similarly, the description could be transferred, in accordance with a multidomain object policy, (separately or with the component information objects) to another end system (see Section 6).

Before a security context can be created for the activities of a user in a particular information domain, the system must be informed which information domain is to be used. Ordinarily, the user's identity must be obtained and authenticated to determine if the user is a member of the requested information domain. One way of performing this startup function is to create a "login" security context that represents one of the end system information domains. The activities allowed in the login security context are limited to authenticating the user identity and starting a security context for the requested information domain (there might be a default information domain for a user recorded in the end system security management information base).

One useful resource control concept is *type enforcement*. The type enforcement concept generally restricts the input and output of a particular function to be of delineated types. In turn, the functions that are allowed to invoke other functions can be controlled by careful specification of input and output types. It is possible to impose a particular implementation of type enforcement by making specific security-critical functions "members" of particular end system information domains. Thus, only "member functions" of an end system information domain could invoke specific executable end system functions.

A consequence of the strict isolation aspects of the end system architecture is that many aspects of covert channels, both timing and storage, either cease to be concerns or are easily controlled. Possible storage channels are reduced to those between security contexts. If information domain policies are properly stated and the security policy, strict isolation, and interprocess communications functions are performing properly, there will be no covert storage channels

available. To exploit timing channels between security contexts requires that a complete security context list is available so that a user can determine which security contexts (including end system security contexts) are in operation. Such information is part of one or more management information domains. It is not likely, and certainly not necessary, that an arbitrary user would be able to access such information. Even for those security contexts in which management information is available to its users, timing information for other security contexts should not be made available to those users.

4.2.3 Security-Critical Functions

The security-critical functions described in this section implement the various security services allocated to the end system and several additional supporting services. This set of security-critical functions is not necessarily complete as presented. Experience through prototyping and experimentation is needed to guide implementations that will meet all of the DGSA requirements, but the functions presented below should provide a sufficient basis for further research.

4.2.3.1 Security Policy Decision Function (SPDF)

The separation of security mechanisms from security policy enforcement and decisions is crucial to the flexibility of the end system security architecture. The SPDF is responsible for making all security policy decisions. The primary role of the SPDF is to isolate the rest of the end system software from knowledge of security policies. The importance of this approach is threefold.

First, the support of multiple information domains with different policies is accomplished easily because the security policies are represented in only one place and are interpreted by only one function. In many current secure system designs, it is difficult to point to the actual software code that implements the single security policy of those systems because it is embedded and scattered throughout code that performs multiple functions.

Second, by keeping security policy representations in one place, it is relatively easy to install, modify, or even replace the security policy for an information domain. It is not necessary to rewrite trusted software that implements the security policy. Rather, the rules that the SPDF interprets for an information domain are updated or replaced.

Third, changing the implementation of the SPDF would be transparent to the operation of the remainder of the end system software. Any correct implementation of the SPDF is acceptable, but it may be useful to standardize the representation of security attributes and security policy rules.

The SPDF approach will allow security-critical functions to be implemented independently of particular security policies. There is the potential in this approach that a computer vendor could support its entire customer base within a single end system software design. To illustrate this concept, consider an example of three enterprises with different, or even conflicting, security policies. The first is a DoD organization using a conventional DoD security policy. The second is a corporation with requirements for data integrity and data separation based solely on need-to-

know authorization. The third is a university research laboratory that does not have any special security needs except a basic privacy-based access control policy. Without a policy-independent architecture, these three differing security policies would result in three different operating system implementations that could cause serious compatibility problems for a vendor trying to support all three environments. Using the SPDF approach, any or all of the three policies could be supported by the same end system software. If necessary, the three enterprises could be served by the same end system or (using the transfer system) they could share information as necessary across different end systems.

4.2.3.2 Authentication Function

The authentication function invokes one or more mechanisms used by an end system to identify and authenticate users (and to authenticate an end system to users), and for end systems to authenticate one another in a distributed environment. A common interface to the authentication function is used that is independent of the any information domain security policy or the authentication mechanisms employed. That is, the authentication function is the service interface to the mechanisms used to identify and authenticate users and end systems. The exact mechanisms selected will depend on the information domain policies in effect. An end system supporting multiple information domain policies may need to implement more than one authentication mechanism.

An authenticated user identity may be passed between information systems rather than the information used to authenticate that identity. That is, an end system supporting a particular information domain would be expected to accept that the authentication function has been performed reliably and correctly by other end systems supporting that information domain (use of the absolute protection concept makes this assumption reasonable). In some cases, it may be necessary to pass information about the authentication mechanisms used to validate the user identity. The transfer system is expected to protect the authenticated user identity as it is passed between information domains. Additional detail about distributed end system interactions is given in Section 7.

4.2.3.3 Audit Function

The audit function accepts audit messages from functions in the end system in accord with information domain and management information domain security policies. Audit records may become part of the security management information that is part of an information management domain (for one or more information domains or end system domains). Audit records may be directed to multiple repositories. In some cases, the audit information may best be used by an individual user (for example, time and method of most recent end system or information domain use). The audit function guarantees that audit messages cannot be lost and that the ordering of messages is preserved. As part of a distributed audit system, audit functions can forward the audit data they collect to a base-level, regional, or central audit center to alleviate local audit data storage requirements and to coordinate audit information from different end systems or LSEs. Audit data must be protected from unauthorized access or modification.

4.2.3.4 Process Scheduling Function

In operating systems that share the end system processor among multiple processes, the process scheduling function determines which of the processes next uses the processor (or processors in a multiprocessor end system) and for how long. The process scheduling function must be included among the security-critical functions so that no process can deny the processor to other processes either purposefully or inadvertently.

4.2.3.5 Device Management Functions and Device Controllers

The remainder of the security-critical functions are each responsible for a particular class of end system resources described below. These resources include memory, storage devices, display systems, interprocess communications, cryptographic services, and any other input/output devices controlled by the end system.

- The *memory management function* is responsible for controlling the use of memory by all software, including security-critical functions. It maintains memory-mapping information and controls the hardware functions that perform memory mapping.

- The *file management function* is responsible for controlling the use of storage media devices. Like the memory management function, it maintains disk-mapping (or other media-specific) information that provides basic virtualizations of the actual storage media. Other software (e.g., database programs) may build upon these virtualizations to provide even more abstract file structures to applications and users.

- The *display management function* is responsible for controlling the use of display devices (including screens and printers), keyboard devices, and pointing devices (e.g., trackballs, mice). The display management function provides basic display device operations. Because a single display device may be used to present information from multiple domains at the same time (typically through multiple windows or on paper), the display management function must maintain information that associates particular information to be displayed with the appropriate security context. Other software (e.g., an X Window System implementation) may provide requests to the display management function to achieve a particular display format.

- The *interprocess communications management function* is responsible for controlling the interprocess communications mechanisms (e.g., locks, semaphores, messages) used by all software processes in the end system. In particular, inter-context (e.g., inter-information domain) transfers are carried out through this function.

- The *cryptographic services management function* is responsible for controlling all of the cryptographically based security mechanisms in an end system. The security services it may support include confidentiality, data integrity, data origin authentication, and non-repudiation. The cryptographic management function may control a number of alternative cryptographic mechanisms to support different services and to provide different levels of protection that satisfy different security policies. The choice of mechanism may be based on

many factors including the sensitivity of the data being protected, the security service requested, and the mechanisms available on other end systems for data that will be transferred.

- Each of the physical devices in the end system, including memory, disks and other storage devices, displays, cryptographic engines, specific user authentication devices, and communications interface controllers, has a corresponding software program that controls and passes information to and from it. These software programs collectively are called *device drivers*. Every device driver must be considered security critical because this software ultimately determines how a device operates. Although device drivers in older end system platforms were often quite large and complex, many contemporary devices contain much of the former device driver function in the device logic or in their own programs. Thus, many device drivers are now reasonably straightforward and follow well-known paradigms, which make their evaluation easier, although great reliance is placed on the correct implementation of the device.

4.2.4 Security-Related Functions

Some software functions within the end system are required to manage information or to provide an interface to the security-critical functions, but are not critical to system security. Of particular interest here are residual operating system functions, security management functions, and transfer system functions.

4.2.4.1 Residual Operating System Structure

Most of the security-critical functions are part of traditional operating system structures. Many other operating system components are not included in the security-critical functions, such as the user interface, utility functions, and high-level abstractions of information. These functions are present in varying forms in all traditional operating systems. The user interface, the particular utility functions, and the information abstractions provided characterize a particular operating system. That is, they distinguish one operating system from another even though they provide essentially the same services to a user. Because the security-critical functions provide commonly used, low-level services, many different operating systems can be implemented using them. Figure 4-2 is an abstract illustration of the software supporting a single security context.

Since security contexts are separated from one another, each can rely upon a different residual operating system structure. Thus, a single end system can support different operating system environments concurrently. Applications that were written to operate with a particular operating system should not require change unless they were allowed to directly manipulate basic operating system functions now controlled by security-critical functions.

Existing operating system implementations will need to be modified to use the standard kernel interface and the services provided by the security-critical functions. The degree of difficulty in making these modifications will be reduced if the original operating system implementation was

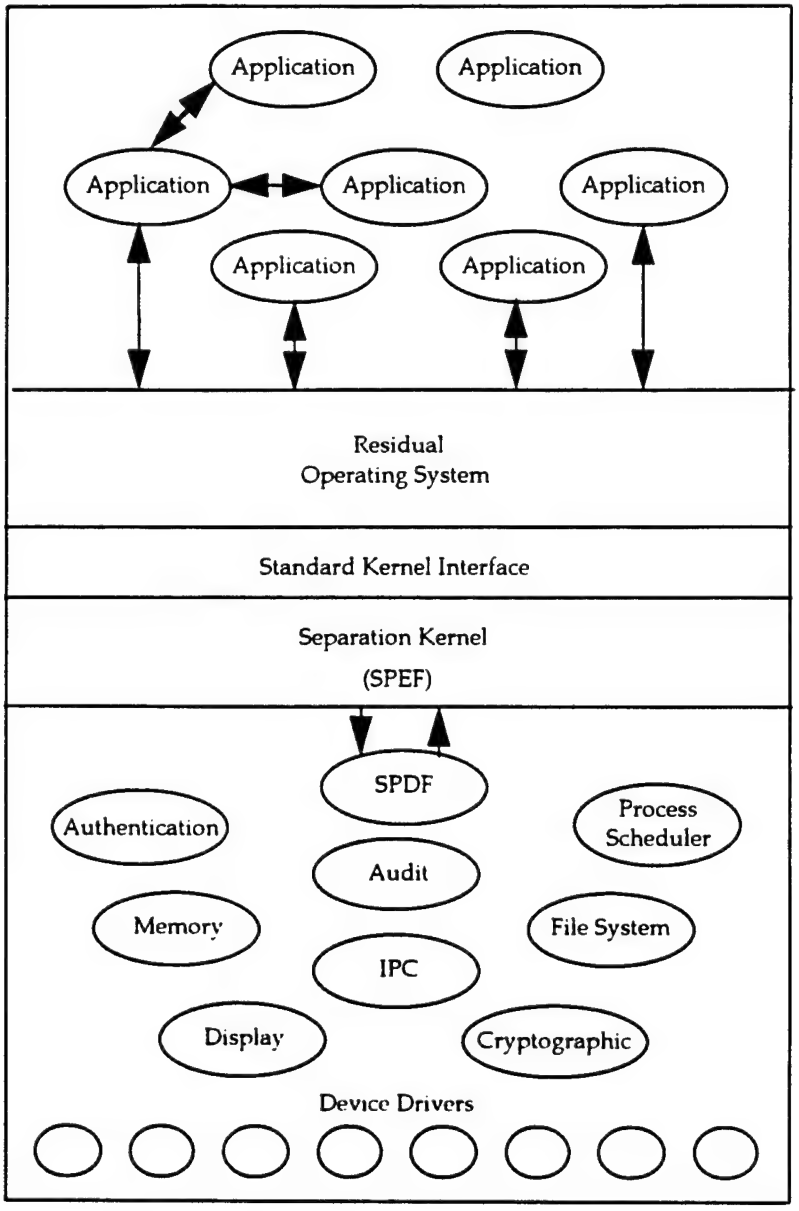

Figure 4-2. Security Context Software Component Relationships

well structured and modular. Some existing secure operating system implementations will adapt relatively easily to the use of the standard kernel interface, and many of the security-critical functions will already be present.

Residual operating system implementations structured to use the standard kernel interface to obtain basic services should be able to be moved among different hardware bases relatively easily since most hardware dependencies will be visible only in the separation kernel and the device drivers. This technique will enable applications to be used even in the face of changing hardware systems.

It should be noted that existing trusted software subsystems (e.g., trusted database applications) also will need to be restructured to fit the end system security architecture. It is possible that such a subsystem might be written to make direct use of the standard kernel interface (rather than calling on the residual operating system) for reasons of efficiency. It also is possible that existing trusted applications (which are appropriately structured) that run on dedicated servers may be able to support multiple information domains through carefully constructed interfaces.

4.2.4.2 Security Management Function

The primary role of the security management function is to control information needed by security-critical and security-related functions within the end system security architecture. Security management is a particular instance of general management functions. The concepts and structures defined in ISO 7498-2 and ISO 7498-4, have been adopted for use in the DGSA. Examples of the information manipulated by the security management function include information domain security policy rules used by the SPDF, configuration parameters for security mechanisms (e.g., cryptographic algorithms), configuration parameters for cryptographic mechanisms and end system devices, and audit information. Some information is managed for specific information domains and some is managed for end systems or LSEs. Details on security management are contained in Section 5.

4.2.4.3 Transfer System Function

The transfer system is defined in accordance with ISO 7498-1 and ISO 7498-2. Communications applications (e.g., X.400 electronic mail (International Telegraph and Telephone Consultative Committee (CCITT), 1988), X.500 directory services (CCITT, 1992), file transfer) and communications protocols used to communicate with other end systems are implemented as untrusted applications within the end system security architecture. These applications make requests for security services (which process information and generate protocol information) that provide required protection. For information to be transferred between end systems and within an information domain, a distributed security context is established through the use of security management and transfer system applications, and security-critical functions. Details of the transfer system are presented in Section 7.

4.3 END SYSTEM SECURITY ARCHITECTURE TECHNOLOGIES

Technologies that are considered to implement the end system security architecture affect all of the elements identified in Section 4.1 (local subscriber environment, hardware, and software).

4.3.1 LSE

The allocation of security services to the LSE requires that mechanisms must be in place to support those services. Physical and administrative security mechanisms will be used to implement the LSE protections. Some areas and mechanisms identified for additional investigation and research are biophysical (e.g., authentication, physical access control) and electronic (e.g., physical access control).

4.3.2 Hardware

The allocation of security services to the hardware requires that mechanisms must be in place to support those services. Some areas and mechanisms for additional investigation and research, and the security services they support, are listed below:

- Fault Tolerance - availability
- Fault Detection - availability, integrity
- Memory Management - strict isolation, integrity
- Protected Mode/Multistate Processors - strict isolation, integrity
- Majority Logic - availability, integrity
- Multiprocessor Architectures - availability, strict isolation, integrity
- TEMPEST - confidentiality
- QUADRANT - availability, integrity.

4.3.3 Software

The allocation of security services and other security-critical functions to the software requires that mechanisms must be in place to support those services. Some areas and mechanisms identified for additional investigation and research are listed below:

- Separation kernels - strict isolation and access control
  - Separation kernel interfaces
  - Process subsystems
    - -- Interprocess communications
    - -- Buffer caches
    - -- Security policy enforcement functions
- Security-critical functions - authentication, confidentiality, integrity, access control, non-repudiation
  - Security policy decision functions
  - Audit functions
  - Cryptographic engine functions

- Device drivers
- Window managers

- Security-relevant functions
  - Security management functions
  - Transfer system functions

- Trusted applications
  - Databases
  - X Window System
  - Operating systems.

5.0 SECURITY MANAGEMENT

Security management provides supporting services that contribute to the protection of information and resources in open systems in accordance with applicable information domain and information system security policies. This section builds on the definitions and concepts presented in Section 3.3. In Section 5.1, critical aspects of security management are related to architectural elements and concepts of the DGSA. In Section 5.2, clause 8 of ISO 7498-2 is used as the basis for presenting details of the DGSA security management architecture. Section 5.3 identifies tools needed by security architects and security administrators, and Section 5.4 discusses standards needed to support DGSA security management.

Security management is a particular instance of information system management. *Managed objects* are information system resources that may be managed. *Management information* is information associated with a managed object that is operated upon to manage that object. A human administrator employs a *management application process* (MAP) to use and maintain management information contained in a logical repository called a *management information base* (MIB). The contents of a single logical MIB may exist in several LSEs. When it is necessary to refer specifically to the processes and management information for security management, the terms *security MAP* (SMAP) and *security MIB* (SMIB) will be used. Otherwise, statements applying to MAPs and MIBs are understood to apply to SMAPs and SMIBs as well.

To ensure efficient and flexible system management, it is generally required that administrators have local or remote access to MIBs. As a result, MAPs will exist in all LSEs. CNs also contain MAPs and MIBs associated with their management. LSEs will manage their LCSs and also may need to cooperate with CN management. In most instances, this cooperation will not involve the use of security-related information since there are no shared security responsibilities.

Since management information comprises specially designated sets of information objects, these sets must exist within an information domain. Several possible choices can be made concerning the information domain in which particular management information objects exist relative to the information domain being managed:

- Each information domain may have a corresponding management information domain (1:1).
- A single management information domain may contain the management information objects for several information domains (1:many).
- The management information objects may be part of the information domain (embedded).

The first two choices are appropriate when the SMIB should not be contained in the information domain to be managed. The last choice, in which the MIB is a part of the information domain being managed, implies that every member of the information domain has the same access privileges to the MIB as to any other information objects in the information domain.

In addition, some management information objects may be associated with an entire information system and its functions. The system MIB might exist in its own management information domain or it might be placed in another management information domain (the latter situation is most likely when a "1:many" management information domain relationship is used).

5.1 SECURITY MANAGEMENT RELATIONSHIPS TO DGSA CONCEPTS

The requirement to manage multiple information domains has the most significant impact on traditional approaches to security management. Traditional security management is based on the assumption that all users of an end system are subject to the same security policy, so that a single view of security management is sufficient for the entire end system. End systems that support multiple information domains must provide the ability to manage each information domain independently. In addition, the use of security services and security mechanisms shared among multiple information domains requires security management coordination at the end system level. Thus, an end system security policy is necessary to specify how the shared use of security functions and resources among information domains is accomplished. This end system policy also must be managed.

As a result of this focus on security policy management, DGSA security management is mission driven and information oriented because information domains are the reflection of mission decisions on how to organize and control information. Section 2 discussed the relationships among missions, requirements, security policies, and security architectures, but only to the granularity of the entire mission. Information domains typically will reflect a major mission function, so further refinement of the mission-specific security policy into an information domain security policy will be necessary. It is not appropriate to specify exactly how that refinement should be done since only general guidelines exist for creating an information domain. However, a number of elements of information domain and end system security policies will be typical for a wide range of mission functions. Several of these security management elements of security policy are listed below, but the lists are not all-inclusive. Section 8 includes examples of incorporating security management policy elements into information domain and end system security policies.

A typical information domain security policy might include some or all of the types of information listed below. Not all of these information types will be reflected in the information domain security policy rules interpreted by the SPDF in an end system, but they are necessary to the development of those rules. Security management in end systems is concerned with the installation, maintenance, and enforcement of these rules and the information about users, security services, and security mechanisms needed to achieve a security policy. Not all security management activities are performed in end systems and relay systems. There are always supporting security management activities that are related to administrative and environmental security mechanisms or which are prerequisite to the use of end system security management functions (e.g., issuance of physical credentials to users, hiring and scheduling human guard services, or carrying out routine maintenance of physical barriers). Although these supporting

activities are not called out in most parts of Section 5, they must be understood to be an integral part of security management. Examples of information domain security policy elements include:

- A brief description of the mission area and a more comprehensive description of the specific mission area function that the information domain supports
- A description of the information objects and their attributes, including rules pertaining to creation and use of multidomain information objects
- Membership criteria
- Rules for interdomain transfers, if any
- Security service requirements (including strength of service) appropriate to meet the risks determined by a threat analysis. Security services should be allocated to LSEs, end systems and relay systems, and the transfer system
- Criteria for acceptable security mechanisms to implement the required security services
- Security management-specific requirements
    - Relationship of the security management information domain to an information domain (1:1, 1:many, or embedded)
    - Criteria for security administrators (e.g., must be a member of the information domain, must not be a member of the information domain)
    - Roles, privileges, and duties of security administrators
    - Identities of security administrators
    - Configuration management requirements for the establishment or modification of information domain security policy rules
- Identification of one or more members of the information domain who are responsible for accrediting information systems that will support the information domain.

The security policy for an end system that supports multiple information domains must specify the management rules for conducting the following activities:

- Providing strict isolation among information domains
- Invoking and managing security mechanisms that implement the security services required by the security policies of the individual information domains
- Developing rules for the management of multidomain information objects, including criteria for user access, display labeling, and transfers between end systems

- Controlling and maintaining security management mechanisms and information objects that enable a security manager of a particular information domain to control that information domain independently of others.

The security policy rules for both end system security management and information domain security management are part of their SMIBs. For an information domain that is supported in more than one end system, the security administrator may have physical access to only some of those end systems. Thus, the SMAP that operates on the portion of a SMIB in a particular end system must be accessible to the security administrator both locally and remotely. A SMAP is like any other application in that it operates in a security context which represents a security administrator (or process) operating in a particular security management information domain. Thus, its security policy is interpreted and enforced by the SPDF and SPEF and it is subject to the same strict separation mechanisms as other information domains.

5.2 ISO 7498-2 AND DGSA SECURITY MANAGEMENT CONCEPTS

Clause 8 of ISO 7498-2 addresses many aspects of security management for open systems interconnection. The ISO 7498-2 security management structure is adopted as the basis for the DGSA security architecture and is extended to apply to all aspects of open systems security management.

5.2.1 Information Domains

ISO 7498-2 begins its security management discussion by considering security policy and security domains (clause 8.1.2):

> There can be many security policies imposed by the administration(s) of distributed open systems and OSI security management standards should support such policies. Entities that are subject to a single security policy, administered by a single authority, are sometimes collected into what has been called a "security domain". Security domains and their interactions are an important area for future extensions.

In the DGSA, "information domain" is substituted for "security domain." Some of the future extensions noted above have been included in the OSI Security Frameworks Overview, ISO 10181-1 (ISO, 1995c). The Frameworks Overview allows, but does not require, security domains to have subset and superset relationships. The DGSA does not allow information domains to be hierarchically related, and so has no need for the subset and superset notions. When sensitivity of information objects is a part of an information domain security policy, all the information objects in an information domain have the same sensitivity. The sensitivity of an information object is a consequence of its presence in an information domain. The "single authority" is the membership of an information domain. Usually the authority is delegated to one or more security administrators for day-to-day security management activities. The reference to "security domain...interactions" is accounted for in the DGSA by security policy interdomain transfer rules and their implementation.

5.2.2 Security Management Information Bases

ISO 7498-2 (clause 8.1.4) describes security management information bases as follows:

> The Security Management Information Base (SMIB) is the conceptual repository for all security-relevant information needed by open systems. This concept does not suggest any form for the storage of the information or its implementation. However, each end system must contain the necessary local information to enable it to enforce an appropriate security policy. The SMIB is a distributed information base to the extent that it is necessary to enforce a consistent security policy in a (logical or physical) grouping of end systems. In practice, parts of the SMIB may or may not be integrated with the MIB.

The DGSA uses SMIBs to conduct information domain and end system management, rather than for only end system management as implied above by the "appropriate security policy" for "each end system." As noted earlier, a distinct security management information domain may be responsible for the management of a single information domain (1:1) or several information domains (1:many), or the information domain may contain its security management information domain (embedded). The SMIB in these cases, respectively, contains security information for the single information domain, contains security information for all of the several information domains, or is contained in the information domain with its information objects. In the 1:many case, the information domains may or may not be related to the same mission. This flexibility allows a security administrator (or group of security administrators) to manage more than one information domain from the same SMIB. Also, it implies that each security administrator has the same attributes (privileges) with respect to the management information of all of the information domains that share a management information domain. (However, not every security administrator necessarily has the same attributes as the other security administrators.)

5.2.2.1 Information Domain SMIB Content

The following examples of information objects might be placed in a SMIB to manage an information domain:

- Information domain security policy rules
- Member registration information
- Member authentication criteria (e.g., strength of mechanism required)
- Member authentication information
- Member attributes (privileges) (e.g., access privileges, release authority for interdomain transfers)
- Visible security label information (i.e., what label, if any, is attached to information that is printed or displayed)

- Security service and security mechanism requirements for specific applications, including intradomain communications and interdomain information transfer.

5.2.2.2 End System SMIB Content

The end system SMIB contains information for management of security functions and resources shared by several information domains, including hardware resources, security-critical functions (particularly security services and mechanisms), and supporting applications (e.g., key management). More detail is given in later sections on several of the supporting security applications and related functions. The following example classes of information objects might be included in the end system SMIB:

- End system security policy rules
- Security services management information (see Section 5.2.7)
- Security mechanisms management information (see Section 5.2.8)
- Supporting services and mechanisms management information (e.g., alarm reporting, information system auditing, cryptographic key distribution, security contexts, security-critical functions, security-related applications operating for the end system).

5.2.3 Communication of Security Management Information

ISO 7498-2 (clause 8.1.5) observes the following about the communication of security management information:

> Management protocols, especially security management protocols, and the communication channels carrying the management information, are potentially vulnerable. Particular care must therefore be taken to ensure that the management protocols and information are protected such that the security protection provided for usual instances of communication is not weakened.

Security management information will be protected in accordance with the security policy of each management information domain. Management applications used to communicate security management information will rely upon the same open system protocol infrastructure as other applications. Management applications operate in security contexts. Security associations that ensure secure communications between security contexts in different end systems are described in Section 6.

5.2.4 Distributed Security Management Administration

ISO 7498-2 (clause 8.1.6) describes distributed security management administration:

> Security management may require the exchange of security-relevant information between various system administrations, in order that the SMIB can be established or extended. In some cases, the security-relevant information will be passed through non-OSI communication paths, and the local systems administrators will update the SMIB through methods not standardized by OSI. In other cases, it may be desirable to exchange such information over an OSI communication path in which case the information will be passed between two security management applications running in the real open systems. The security management application will use the communicated information to update the SMIB. Such updating of the SMIB may require the prior authorization of the appropriate security administrator.

The DGSA is consistent with this view and uses it as the basis for DGSA distributed security management. Each management information domain uses and maintains the SMIB for the information domain it manages. Security administrators may rely on a custodial infrastructure (e.g., communications security custodians). Cooperation with local administrators may be necessary for functions that cannot be managed remotely (e.g., aspects of key management that require physical access and personal accountability dictated by administrative and environmental considerations).

5.2.5 Security Management Application Protocols

ISO 7498-2 (clause 8.1.7) requires security management application protocols for exchange of security-relevant information:

> Application protocols will be defined for the exchange of security-relevant information over OSI communication channels.

There is not yet a clear preference among existing and developing security management application protocols. The general management application protocol defined by ISO is the Common Management Information Protocol (CMIP) (ISO, 1991). ISO also had defined the General Upper Layer Security (GULS) Security Exchange Service Element Protocol (SESEP) (ISO, 1994b). In addition, several security management functions have been defined with the series of standards within ISO 10164.

The Internet Engineering Task Force (IETF) defined the Simple Network Management Protocol (SNMP) (Case, 1989) and its successor, SNMP version 2 (Case, 1991). As the security management protocol situation becomes stable, the DGSA will adopt appropriate protocols.

5.2.6 End System Security Management Functions

ISO 7498-2 (clause 8.2.1) observes the following about system security management:

System security management is concerned with the management of security aspects of the overall OSI environment. The following list is typical of the activities which fall into this category of security management:

a) overall security policy management, including updates and maintenance of consistency;

b) interaction with other OSI management functions;

c) interaction with security service management and security mechanism management;

d) event handling management;

e) security audit management; and

f) security recovery management.

As noted previously, the DGSA broadens the view of end system security management to the entire open systems environment, especially with respect to the support of multiple information domains. The topics of event handling, security audit, and security recovery management are interrelated and will be treated together.

ISO 7498-2 (clause 8.3.1) describes event handling management as follows:

The management aspects of event handling visible in OSI are the remote reporting of apparent attempts to violate system security and the modification of thresholds used to trigger event reporting.

ISO 7498-2 (clause 8.3.2) describes security audit management as follows:

Security audit management may include:

a) the selection of events to be logged and/or remotely collected;

b) the enabling and disabling of audit trail logging of selected events;

c) the remote collection of selected audit records; and,

d) the preparation of security audit reports.

ISO 7498-2 (clause 8.3.3) describes security recovery management as follows:

Security recovery management may include:

a) maintenance of the rules used to react to real or suspected security violations;

b) the remote reporting of apparent violations of system security; and

c) security administrator interactions.

These security functions are related since the event handling function deals with all the apparent security violations recognized by an end system, the audit function selects those events that will be recorded, and the recovery function acts upon some of the selected events. The selection of audited events and those requiring a recovery action is determined by information domain security policies or by the end system security policy.

Event handling includes local as well as remote reporting of security-related events. Depending on whether a management entity (a security manager or a security recovery application) or a user is expected to examine or act on various alarms or audit records, alarm or audit information objects may be recorded in a particular management information domain SMIB, an end system SMIB, or a user-accessible file in an information domain.

Security recovery actions might include terminating a particular security context, temporarily prohibiting certain activities within an information domain, or disabling a particular communications interface. Some security recovery actions may depend on specialized data structures, such as a compromised cryptographic key material list, which controls continued use of key materials.

5.2.7 Security Service Management

ISO 7498-2 (clause 8.2.2) describes security service management as follows:

> Security service management is concerned with the management of particular security services. The following list is typical of the activities which may be performed in managing a particular security service:
>
> a) determination and assignment of the target security protection for the service;
>
> b) assignment and maintenance of rules for the selection (where alternatives exist) of the specific security mechanism to be employed to provide the requested security service;
>
> c) negotiation (locally and remotely) of available security mechanisms which require prior management agreement;
>
> d) invocation of specific security mechanisms via the appropriate security mechanism function, e.g., for the provision of administratively-imposed security services; and
>
> e) interaction with other security service management functions and security mechanism management functions.

An information domain security policy may be very specific about how security service requirements are to be met (by mandating particular security mechanisms). Alternatively, it may give only a general requirement for a security service of a particular strength and allow the SMAP to select an appropriate mechanism from those available. Each of the activities in the list above is concerned with an aspect of determining how security service requirements are satisfied by security mechanisms, as discussed below.

5.2.7.1 Determining and Assigning Strength of Service

Determining security services to be used and their strength is one aspect of developing a security policy for an information domain or an end system. The choices made are dependent on threats, vulnerabilities, and acceptable risk. That is, for large classes of information processing activities, a single determination of required security services can be made in advance because the value of the information being protected does not change often or quickly, nor do the vulnerabilities and risk. There are other classes of information activities for which it may be appropriate for a user to choose whether or not to employ a particular security service. For example, within the same information domain, some electronic mail messages may be of an informal or personal nature and not require a non-repudiation service, but other messages may be official business and may be required (by written policy) to employ a non-repudiation service. In cases like this, the user needs a selective means of invoking the security service, but the strength of the service is likely to be predetermined.

5.2.7.2 Assigning and Maintaining Rules for Mechanism Selection

For a given security service, one or more security mechanisms, alone or in combination with others, may be able to implement it. Some security mechanisms may be able to support more than one security service.

One of the aspects of the principle of absolute protection is that the security services chosen within an information domain security policy each have a minimum strength associated with them. Not all the security mechanisms that support a given security service need to be provided within end systems (or relay systems). In particular, the LSE may employ various administrative and environmental security mechanisms that contribute to the provision of one or more security services. As a result, the security mechanisms that support a given security service may be different when protecting information within an end system than when protecting information between end systems within the same LSE or between end systems in different LSEs. The resulting security service implementations must provide at least the minimum protection demanded by the security policy in all situations. Thus, to the extent that an end system supports security services with different mechanisms and a SMAP is aware (or can be made aware) of the distinctions among activities within an end system, between end systems in the same LSE, and between end systems in different LSEs, alternate choices of security mechanisms could be made.

The added complexity involved in making such choices might lead information system security architects to use only one set of mechanisms that satisfies an information domain security policy in all cases. However, in some situations this strategy would not be appropriate. For example, if some end systems in the same LSE often exchange large files, but only infrequently with end systems in different LSEs, a confidentiality mechanism necessary in the latter case might introduce an unacceptable performance penalty in the local situation, but administrative and environmental mechanisms could be relied upon to achieve the required level of protection.

5.2.7.3 Negotiating Available Security Mechanisms

One or more end systems that support the same information domain may be able to support a particular security service with more than one security mechanism, but it may not be known in advance of attempted communications which of these security mechanisms may be implemented in a specific end system. In such cases, the specific security mechanisms to be employed must be negotiated between the SMAPs in the end systems at the time the security association is established between them.

5.2.7.4 Invoking Security Mechanisms

The invocation of security services and security mechanisms within the end system security architecture involves several functions. Since all security services are security-critical, they are accessible only within the separation kernel, and applications can invoke them only through the standard kernel interface. Since most applications will rely upon the residual operating system for use of the standard kernel interface, the use of the interface will be transparent to those applications. If a request for a security service does not specify a security mechanism, the SMAP makes a choice among the available security mechanisms based on the information domain policy and invokes it through an appropriate operating system call. Otherwise, the SMAP invokes the specified security mechanism.

Although each application could make requests for security services and security mechanisms directly to the SMAP, there are significant advantages to adopting an Application Program Interface (API) approach. APIs provide a common set of subroutine calls to a related set of programming functions or services. An API not only relieves application designers of creating a specific set of interfaces, but also allows underlying services to be replaced (by equivalent mechanisms) without affecting the application implementation. Various efforts are defining APIs for the invocation of security mechanisms. One such effort is the General Security Service (GSS) API intended for use with the Internet suite of communications protocols (Linn, 1993). The GSS API and other related APIs could be used to invoke all security functions by making them the standard interfaces to the SMAP (they could be incorporated into the SMAP). GULS provides a standard set of protocol elements that can be used by applications to convey protected information between end systems.

The use of a combination of the GSS API, GULS, SMAPs, and the standard kernel interface can contribute to the independence of security services and security mechanisms and to their transparency to users and applications. This independence allows different security mechanisms to be accommodated at various stages in an end system life cycle, and for end systems to accommodate information domains with different security service requirements.

5.2.7.5 Specifying Interactions Among Security Service and Mechanism Management Functions

The use of some security services depends on the results of others. For example, access control usually employs the output of the authentication service. Required security service interactions

must be expressed in a security policy. Similarly, some security mechanisms are dependent on others or on supporting security functions, for example, key management for cryptographic security mechanisms. These dependencies must be part of the SMIB so the SMAP can invoke the appropriate security mechanisms and functions.

5.2.8 Security Mechanism Management

ISO 7498-2 (clause 8.2.3) describes security mechanism management as follows:

> Security mechanism management is concerned with the management of particular security mechanisms. The following list of security mechanism management functions is typical but not exhaustive:
>
> a) key management;
>
> b) encipherment management;
>
> c) digital signature management;
>
> d) access control management;
>
> e) data integrity management;
>
> f) authentication management;
>
> g) traffic padding management;
>
> h) routing control management; and,
>
> i) notarization management.

The DGSA adopts this list and adds availability management.

5.2.8.1 Key Management

ISO 7498-2 (clause 8.4.1) describes key management as follows:

> Key management may involve:
>
> a) generating suitable keys at intervals commensurate with the level of security required;
>
> b) determining, in accordance with access control requirements, of which entities should receive a copy of each key; and,
>
> c) making available or distributing the keys in a secure manner to entity instances in real open systems.
>
> It is understood that some key management functions will be performed outside the OSI environment. These include the physical distribution of keys by trusted means.

> Exchange of working keys for use during an association is a normal layer protocol function. Selection of working keys may also be accomplished by access to a key distribution center or by pre-distribution via management protocols."

The DGSA relies upon standard key management techniques. Specifically, a Security Association Management Protocol (SAMP) is a necessary part of the transfer system. There are several competing SAMP developments in progress. Among them is the Institute of Electrical and Electronics Engineers (IEEE) 802.10 Standard for Interoperable LAN/MAN Security (SILS) Part 3 (IEEE, 1995), which has recently become the basis for the key management protocol standard being developed in ISO. The IETF is considering several alternative proposals. The DGSA requires a SAMP that will be sufficiently general to support security association establishment as described in Section 6.

There is an evolving key distribution system for U.S. Government use, the Electronic Key Management System (EKMS), from which the majority of U.S. Government cryptographic keying materials are generated and distributed. The EKMS Local Management Device (LMD) is the EKMS presence in LSEs. The EKMS is adopted as part of DGSA guidance. Although this is specific guidance, it is necessary because key management and cryptographic systems are being developed independently by vendors. A potential customer might procure several key management devices just to support a large, base-level LSE, some of which could be based on proprietary security management systems for vendor-specific end systems or LCS security products. These key management systems would almost certainly be incompatible with one another, thus increasing both initial and life-cycle costs, and impeding interoperability. The clear long-term solution is to develop key management and cryptographic products (including the evolving EKMS) based on the forthcoming standards.

5.2.8.2 Encipherment Management

ISO 7498-2 (clause 8.4.2) describes encipherment management as follows:

> Encipherment management may involve:
>
> a) interaction with key management;
>
> b) establishment of cryptographic parameters; and,
>
> c) cryptographic synchronization.
>
> The existence of an encipherment mechanism implies the use of key management and of common ways to reference the cryptographic algorithms.
>
> The degree of discrimination of protection afforded by encipherment is determined by which entities within the OSI environment are independently keyed. This is in turn determined, in general, by the security architecture and specifically by the key management mechanism.
>
> A common reference for cryptographic algorithms can be obtained by using a register for cryptographic algorithms or by prior agreements between entities.

It is expected that new cryptographic products will support multiple algorithms that can be selected by each application. In such an environment, the registration of cryptographic algorithms will be necessary so that algorithm selection can be negotiated between end systems. The ability to select a cryptographic algorithm has implications for the security management of the devices involved, such as determining under what conditions an algorithm can be employed and for auditing algorithm use.

5.2.8.3 Digital Signature Management

ISO 7498-2 (clause 8.4.3) describes digital signature management as follows:

> Digital signature management may involve:
>
> a) interaction with key management;
>
> b) establishment of cryptographic parameters and algorithms; and
>
> c) use of protocol between communicating entities and possibly a third party.
>
> Note: Generally, there exist strong similarities between digital signature management and encipherment management.

When digital signatures support a non-repudiation service that relies upon a trusted third party, additional security management responsibilities may be added with respect to long-term archiving of keys and algorithm identifiers so that transactions can be verified well after they occur.

5.2.8.4 Access Control Management

ISO 7498-2 (clause 8.4.4) describes access control management as follows:

> Access control management may involve distribution of security attributes (including passwords) or updates to access control lists or capabilities lists. It may also involve the use of a protocol between communication entities and other entities providing access control services.

The "distribution of security attributes" includes their initial installation in a SMIB. Since not all the information in an information domain SMIB is necessarily locally present in every end system that supports an information domain, it may be necessary to convey access control attributes between end systems. Note that user-specific access control attributes may not always be required since an information domain security policy may confer certain access rights on all its members.

5.2.8.5 Data Integrity Management

ISO 7498-2 (clause 8.4.5) describes data integrity management as follows:

Data integrity management may involve:

a) interaction with key management;

b) establishment of cryptographic parameters and algorithms; and,

c) use of protocol between communicating entities.

When using cryptographic techniques to support the data integrity service, similarities exist between data integrity management and encipherment management. In some instances, within a single end system, data integrity can be attained as a by-product of strong access control mechanisms. When a strong communications data integrity service is required, cryptographic mechanisms are likely candidates. A SAMP must provide means for selecting algorithms and keys for data integrity.

5.2.8.6 Authentication Management

ISO 7498-2 (clause 8.4.6) describes authentication management as follows:

> Authentication management may involve distribution of descriptive information, passwords or keys (using key management) to entities required to perform authentication. It may also involve use of a protocol between communicating entities and other entities providing authentication services.

Authentication mechanisms rely upon particular authentication information to validate a given identity. The authentication information against which user-supplied authentication information is verified is stored in the SMIB and is subject to similar considerations as access control attributes. It should be noted that an authenticated individual identity may not be required by some information domain policies since it may be sufficient that an individual has been physically identified and allowed access to an end system to assert membership in an information domain.

5.2.8.7 Traffic Padding Management

ISO 7498-2 (clause 8.4.7) describes traffic padding management as follows:

> Traffic padding management may include maintenance of the rules to be used for traffic padding. For example, this may include:
>
> a) pre-specified data rates;
>
> b) specifying random data rates;
>
> c) specifying message characteristics such as length; and
>
> d) variation of the specification, possibly in accordance with time of day and/or calendar.

Traffic padding in physical layer communications devices is often managed as a configuration parameter. In an open systems environment, traffic padding in the physical layer will occur infrequently. Traffic padding in application layer protocols could be invoked as the result of a user request or as the result of an information domain security policy requirement applied to all or some class of communications. The critical management aspect of satisfying such a request is to assure that the padding is applied at the correct stage of processing with respect to other security services, such as data integrity or data confidentiality.

5.2.8.8 Routing Control Management

ISO 7498-2 (clause 8.4.8) defines routing control management as follows.

> Routing control management may involve the definition of the links or sub-networks which are considered to be either secured or trusted with respect to particular criteria.

Routing control in open systems meeting DGSA requirements will normally be restricted to choosing a particular network interface when an end system is connected to multiple CNs or LCSs.

5.2.8.9 Notarization Management

ISO 7498-2 (clause 8.4.2) defines notarization management as follows.

> Notarization management may include:
>
> a) the distribution of information about notaries;
>
> b) the use of a protocol between a notary and the communicating entities; and
>
> c) interaction with notaries.

See Section 5.2.8.3.

5.2.8.10 Availability Management

Availability management is not described in ISO 7498-2. Availability management is limited to interactions with the LCS- or CN-provided management facilities for notifications of outages and, if applicable, alternate service information.

5.3 SECURITY MANAGEMENT TOOLS

Security architects will need various tools to enable them to design end systems that will support user requirements as reflected in information domain security policies. Security administrators must have available a set of tools to assist them in performing their functions efficiently and conveniently. Not all of the tools discussed here are available currently, and steps will need to be taken to ensure their timely creation.

5.3.1 Security Policy Rule Specification

To complement the development of the SPDF, a tool must be developed to assist in or perform the reduction of security policies to security policy rules that can be interpreted by the SPDF. The specification of security policy rules is a new endeavor and will require a significant research effort.

5.3.2 Security Mechanisms Catalog

The selection of appropriate security mechanisms to implement the security services required by security policies is an activity that will require specific support that does not yet exist. There are several interrelated factors that must be considered.

The first factor is the strength of security mechanisms and other security-critical functions (e.g., separation kernel effectiveness). The second factor is the characteristics of security mechanisms, that is, what they do and do not provide, how security mechanisms interact with one another, and implementation and employment requirements for security mechanisms to work effectively. The third factor is the cost of security mechanisms, including both procurement and life-cycle costs (to include supporting functions such as key distribution). The fourth factor is user impacts, such as performance penalties.

To an extent, some of these factors are considered in current procedures for evaluating security products. To support security architects in suggesting appropriate security mechanism choices, all of these factors must be considered. Evaluations based on these factors could be performed on implementations of particular security mechanisms or on products that implement multiple security mechanisms. The result of such evaluations would be a security mechanisms and product catalog from which security architects could make appropriate choices.

One significant aspect of the evaluations for such a catalog is that they would not result in a single composite rating for a security mechanism or product. Each security mechanism would be rated for its strength in support of a particular security service. A security mechanism that supports more than one security service would have more than one strength rating. The security mechanism might have a different strength rating when used in conjunction with one security mechanism than it would with another. A security product would have strength ratings for each of its mechanisms. Clearly, establishing metrics for these strength ratings will be a formidable and critical aspect of creating the catalog.

5.3.3 Maintenance Applications for Security Administrators

Each of the security management activities discussed in Section 5.2 will require automated support for security administrators. The applications that provide this support are concerned with various aspects of SMIB maintenance, key management, and examination, processing, and correlation of information such as audit records. These management applications should work together smoothly, but they must also be separable if it is desired to assign certain activities to specific security administrators. In some instances, it will be necessary to integrate security management applications with other applications. For example, X.500 Directory Service Agents

might be used to store portions of a SMIB so that user certificates are easily available to a user community.

5.4 AREAS FOR SECURITY MANAGEMENT STANDARDIZATION

Standardization of security management functions, data structures, and protocols will enable interoperation of SMAPs across many end system platforms and, thus, allow effective distributed security management. Areas for security management standardization include, but are not limited to the following:

- Security policy rule representations so that security policies can be installed remotely
- Key management functions that support the generation, distribution, and accounting of cryptographic key material
- Audit information formats so security management applications can interpret events occurring on multiple end systems that support multiple security domains
- Protocols for the exchange of security management information and for remote security management operations.

6.0 TRANSFER SYSTEM

This section discusses the basic goal of the transfer system security architecture and then the means to achieve that goal. Section 6.1 discusses the basic notion of distributed security contexts and the primary function that supports them, the security association. Section 6.2 describes several supporting functions and tools needed to implement distributed security contexts and security associations. Section 6.3 discusses the relationship of the transfer system security architecture to some specific security-related topics.

In Section 3, the transfer system was identified as the LCSs, CNs, and the communications protocols in end systems and relay systems. Security services allocated to the transfer system provide the basis for the protection of information in transfer. Availability is the only security service allocated to CNs and LCSs. Additional security services may be provided by LCSs, but they are only applicable to local communications.

The portion of the transfer system in end systems and relay systems consists of open system networking applications and communications protocols (including some security protocols). These applications and protocols are executed in the same security context as other user applications for a user operating in a particular information domain. Except for transfer system functions that are among the security-critical functions (e.g., network interface device drivers, cryptographic functions), transfer system software does not need to be trusted. The transfer system must be managed, so the SMAP and SMIB of Section 6 are extended to account for transfer system functions.

The primary goal of the transfer system security architecture is to provide protection of information in transfer to support information sharing and distributed processing within the security architectures of the other DGSA elements and the fundamental concepts. The basic approach to achieving this goal is to enable security contexts in different end systems or relay systems (that support the same information domain) to communicate as if they were in the same end system or relay system. The transfer system security architecture must fit within the end system and relay system architecture of Section 4 and the security management architecture of Section 5, and it must extend the support of fundamental DGSA concepts to communications, especially information domains, strict isolation, multidomain information objects, and absolute protection. The remainder of Section 6 addresses various concepts and functions needed for achieving the transfer system goal.

6.1 DISTRIBUTED SECURITY CONTEXTS

The generic transfer system security architecture seeks to create structures in which applications in security contexts in different end systems or relay systems (that support the same information domain) communicate with the same assurance as if they were in the same end system or relay system. Such structures are referred to as *distributed security contexts*. There are two basic classes of communications that must be considered, *interactive* and *staged delivery*. Staged delivery refers to communications in which the information being transferred is sent from the

originating end system application to a relay system application, in its entirety, and then is sent from the relay system application to the destination end system application. (There may be several relay system applications involved before the information is finally delivered to the destination end system application.) The most common example of staged delivery is electronic mail. Interactive communications include all non-staged delivery applications. The means used to create distributed security contexts are different for interactive and staged delivery communications and will be discussed separately.

6.1.1 Distributed Security Contexts for Interactive Communications

An *interactive distributed security context* is formed when two security contexts in different end systems are joined securely using a set of mechanisms that is referred to as a security association. A *security association* is the totality of communications and security mechanisms and functions (e.g., communications protocols, security protocols, administrative and environmental security mechanisms, security-critical mechanisms and functions) that securely binds together two security contexts in different end systems or relay systems supporting the same information domain[1]. A security association extends the protections required by an information domain security policy within an end system to information in transfer between two end systems and it maintains strict isolation from other information domains. A security association can be considered an extension or expansion of an OSI application association. OSI application layer entities in different end systems employ application associations to communicate. An application association is composed of appropriate application layer functions and protocols plus all of the underlying communications functions and protocols at other layers. A security association is an application association that includes additional support from security functions and mechanisms. The security management information for a security association is contained in a SMIB and includes all the security-relevant attributes required to establish and maintain a security association, such as the information domain label and secure communications attributes (e.g., cryptographic algorithm identifiers and keys).

Making a decision about whether to allow establishment of a security association may require several related functions to be performed such as the exchange and processing of security attributes of the user (e.g., authenticated identity, access privileges). These attributes might be contained in a security certificate such as that defined in the X.509 Directory Services Authentication Framework (CCITT, 1992). The information contained in an X.509 certificate may be signed by any number of hierarchically related certificate-issuing authorities, down to an information domain-specific certificate-issuing authority if that level of granularity is required. This signature verification adds greater assurance to the credibility of the information contained in the certificate.

---

1 Note that the DGSA meanings of security association and security association management protocol are more general than their meanings in existing protocol specifications.

Multiple security protocols may be included in a single security association to provide a combination of security services. For example, a network layer protocol might provide continuous end system origin authentication and data integrity, while a presentation layer protocol might provide selective field data confidentiality. Some lower layer security protocols can multiplex several security associations between the same end systems. The security associations share the same cryptographic algorithm and keys. This arrangement may be appropriate for interactive distributed security contexts that support the same information domain, but it is unlikely to be acceptable for different information domains because of strict isolation requirements.

In some instances, an interactive distributed security context will be formed between end systems that employ no security protocols and may not even require an authenticated user identity. Such instances include access to public information utilities (e.g., a news wire service feed) or completely unprotected end systems. In these instances, an end system that supports other information domains, will be entirely responsible for maintaining the isolation of unprotected information domains from other information domains.

Some communications between end systems involve information that is not ordinarily stored in an end system, for example, real-time voice and video applications. In these cases, users must monitor and enforce the accuracy of the security context and association established for the distributed security context. That is, humans must ensure that information exchanged belongs to the information domain represented by the distributed security context as is currently done when using Secure Telephone Unit-IIIs for secure voice or data communications.

6.1.2 Staged Delivery Distributed Security Contexts

A staged delivery distributed security context is transferred from the originating end system to the destination end system. This is accomplished by an application in the originating end system cryptographically wrapping the information to be transferred in a form that allows the destination end system to reconstitute the security context in which the information was wrapped. The wrapped information is transferred (in stages) from the originating end system to the destination end system. Ideally, the wrapping process should provide all security protection of the information while in transfer. No security services (other than availability) should be expected of the application relay systems involved in the staged delivery because they might be provided by common carrier providers, as is the case for CNs. If the wrapping process cannot provide all the necessary security protection, the application relay systems will have to be implemented to support the DGSA and interactive distributed security contexts between end systems and relay systems will have to be used to ensure the secure staged transfer of information.

There is an existing specification for a secure electronic mail service that satisfies the requirements for staged relay distributed security contexts. This document is the Secure Data Network System (SDNS) Message Security Protocol (MSP) specification (NSA, 1992). For details of how secure staged delivery can be achieved, the MSP specification should be

examined. MSP will be the basis for secure messaging in DoD as Phase II of the DMS is implemented and deployed.

6.1.3 Other Aspects of Distributed Security Contexts

This section provides additional discussion of two specific aspects of distributed security contexts.

6.1.3.1 Multidomain Object Transfer

Section 3.3.1.4 defined and discussed multidomain objects and noted that their purpose is to display or print related information objects from several information domains in an ordered format. Section 3.2.2 discussed some high-level implementation aspects of multidomain objects. The transfer of a multidomain object between end systems requires that both the component information objects and the description of their relationships be transferred. Since a distributed security context supports transfer of information within a single information domain, one distributed security context is used for each of the component information domains. If the description of the component relationships is contained in an information object in a separate information domain, another distributed security context is required for its transfer. An application similar to those used to display or print multidomain objects is needed to coordinate the transfer of the component information objects.

6.1.3.2 Distributed Security Context Single Information Domain Restriction

The definition of a distributed security context restricts it to joining end system or relay system security contexts that support the same information domain. In principle, this restriction could be removed under some conditions for some information domain security policies, however, there are practical reasons for retaining it. One of the principal functions of a distributed security context is to maintain strict isolation of information in transfer. Within an end system, the separation kernel (or other strict isolation mechanism) controls all interactions between security contexts. As noted earlier, it is expected that cryptographic mechanisms will be the usual means to maintain strict isolation for information in transfer. The use of such cryptographic mechanisms requires shared use of keys and other supporting information between security contexts in the communicating end systems. If those security contexts support different information domains, sharing of the keying information is difficult. There will also be additional complexity introduced into many communications and security protocols that will result in trusted implementation of additional functions. The restriction that distributed security contexts support transfers within a single information domain is intended to simplify implementations that support the DGSA concepts.

6.2 TRANSFER SYSTEM SUPPORT

This section describes several elements needed to support the basic transfer system activities.

6.2.1 Security Management Application Process

In addition to the SMAP functions described in Section 5, the SMAP also controls the establishment and termination of all security associations and distributed security contexts, and all transfer system security services and mechanisms. Additional transfer system-related SMAP functions and interfaces support the following activities:

- End system communications applications requests (e.g., through the GSS-API)
- Additional SMIB information object use and maintenance (e.g., to access information for remote security administration maintenance, security protocol and algorithm operation, certificate processing)
- Maintenance and retrieval of security information from the X.500 Directory using the directory access protocol
- MSP processing for staged delivery secure messaging for both transmission and receipt
- SAMP operations for establishment of interactive distributed security contexts, including security protocol operation, termination, and recovery, plus maintenance of SMIB entries for each security association established
- General-purpose management protocol operation (e.g., CMIP) to accomplish secure exchange of security information between distributed SMAPs or network management information requested by network management systems.

6.2.2 Security Management Information Base

Additional information is required in the end system SMIB and the information domain SMIBs to support transfer system operations.

Additional information domain SMIB information items include:

- X.509 certificates to carry appropriate security information, such as key management certificates
- User access control information for distributed operations
- Traffic and message keys
- Accumulated audit data, including records of distributed security context utilization.

Additional end system SMIB information items include:

- Key management, encipherment, integrity, and signature algorithm identifiers, and security protocol objects

- End system access control information for distributed operations
- Encryption algorithm initialization information
- Security association configuration information
- Compromise action information (e.g., revoked certificates lists)
- Contingency plan parameters (e.g., auto-purge and security policy replacement actions under emergency conditions).

Some SMIB items may be held in Directory Service Agents (DSA) for ease of access by many users. Such items might include key management information (e.g., certificates and user keying material). SMIB information stored in X.500 Directories must be integrity protected.

6.2.3 Security Protocols

Several security protocols, either existing or in development, are candidates for use in end systems implementing the DGSA. Others may be added over time.

The Transport Layer Security Protocol (TLSP) is an ISO standard (ISO, 1995b) as is the Network Layer Security Protocol (NLSP) (ISO, 1995a). The IEEE 802.10 SILS Secure Data Exchange (SDE) protocol standard (IEEE, 1992) is appropriate for LCS security services (beyond availability) when needed. MSP is the DoD standard for electronic messaging. The state of SAMP standardization was discussed in Section 5.2.8.1.

6.2.4 Cryptographic Support

The creation of distributed security contexts, which provide communications security services and strict isolation adequate for sensitive information, is usually dependent on cryptographic mechanisms. Thus, the availability of low-cost cryptographic devices is a critical element of the DGSA. These cryptographic devices must be sufficiently flexible to support requirements of different information domains in the same end system.

This flexibility will be achieved if the devices accommodate multiple cryptographic algorithms and multiple key management schemes, including public key encryption schemes and various key distribution center schemes. Otherwise, a multiplicity of cryptographic devices will be needed, resulting in increased costs. To manage these devices, there must be a registry of cryptographic algorithms and key management schemes so that the specific choices can be negotiated for a particular security association.

Currently available cryptographic and key management devices do not meet these flexibility criteria. Very large scale integration (VLSI) chip technology may now have reached a sufficient density to achieve a cost-effective single-chip design which can support multiple algorithms and a variety of key management schemes, along with a cache memory capable of handling reasonable quantities of key material. The cryptographic devices must be capable of a minimum

throughput rate of 10 megabits per second to be useful with high-performance workstations. Isolation techniques must accommodate concurrent algorithm execution. In addition to creating low-cost devices, current custodial functions must be minimized through the use of electronic key management technology.

6.2.5 Distributed Management Systems

Distributed management of information systems both supports the transfer system and relies upon the transfer system for its operation. Management systems will rely upon the same transfer system security structures (distributed security contexts, security associations, and security protocols) as any other application.

When distributed information systems become very large, their management becomes very complex. To make the complexity manageable, hierarchical management approaches are often adopted. It then becomes necessary to coordinate the levels of delegated management authority. The coordination is achieved by the way management information is organized and through the control of that information as required by security policies. Hierarchical management relationships are not reflected in the way management applications communicate with one another. That is, management protocols are peer oriented, not hierarchically related. When the term "hierarchical management system" is used, it must be understood that a set of information relationships is being described, not a communications structure. This means that the hierarchical aspect of management is a human, organizational function. The organizations and administrators that manage information systems may be organized hierarchically. Management information may reflect that organization, but the end systems in which management applications are implemented only communicate as peers.

Management systems are composed of management applications implemented in end systems. Some management applications must coexist with other applications in end systems, but for logistical reasons it may be desirable to dedicate some end systems to management system activities. Management systems can be grouped into three categories based on the particular type of management function being performed. While these categories are logically separate, they often support one another. The three categories are network management, security management, and information management.

Traditional network management systems are network control centers that monitor and configure network components, perform fault isolation functions, and collect accounting and performance information. Security management systems typically provide information to support security services and mechanisms in end systems and relay systems. Most often the support is for cryptographic mechanisms, such as the DoD EKMS. Information management systems include X.500 Directory systems, the Internet Domain Name Service (DNS) and the Network Information Center (NIC).

Although these three logical categories of management systems could be implemented in end systems dedicated to the functions of only one of them, as a practical matter, some of the functions can be expected to be supported on common end systems. However, each logical

category may require unique technical administrative expertise. In some cases, it will not be prudent to assign multiple administrative functions to individuals because too much control might be entrusted to them.

6.3 DGSA TRANSFER SYSTEM ISSUES

Two aspects of the DGSA transfer system deserve further discussion. One is traffic flow security, and the other is potential limitations on distributed processing functions.

6.3.1 Traffic Flow Security in Open System Communications Environments

Full TFS mechanisms are intended to conceal characteristics of communications protocols and information that might be derived from them through unimpeded observation of a communications path. Full TFS mechanisms operate at the physical protocol layer. Only if communications facilities are owned or controlled by user organizations can full TFS be applied. The use of common carrier CNs precludes the use of full TFS mechanisms. One consequence of providing full TFS between two LSEs is that the communications path cannot be used for any other purpose and, thus, creates a closed system.

The clear cost disadvantages of owning and operating private CNs means that there must be a careful examination of threats and vulnerabilities to determine whether full TFS is required. Unless it is necessary to subject all communications to full TFS, the DGSA requirements for open system and common carrier communications can be met with multiple communications connectivity. The strict isolation mechanisms required in end systems make it possible to support multiple communications connections among the information domains supported. Partial TFS mechanisms should be considered as alternatives to full TFS when judged to be appropriate to the known threats and vulnerabilities.

6.3.2 Limitations on Distributed Processing

Some communications technologies are inherently of a broadcast nature (e.g., radio, broadband LANs). Broadcast technologies make it possible to communicate with any end system that has access to the medium without the need to explicitly address information to specific end systems. Broadcast-like effects, called multicasts, can be achieved over non-broadcast communications systems through various methods that address and send information to (possibly large) groups of recipient end systems or users (e.g., groups of electronic mail recipients).

Certain limitations are encountered if cryptographic mechanisms are used to support security services for broadcast (and some multicast) communications. There are two basic choices. First, for true broadcasts, a single encryption key must be shared among all recipients. The use of a shared key among large numbers of recipients not only increases the likelihood that the key will be compromised, but the distribution and use of one or more shared keys is difficult to coordinate. (The same considerations apply to multicast services that depend on broadcast media.)

Second, for multicasts that are addressed to a group of recipients, a single key can be used for the security mechanism applied to the information to be sent and that key can be replicated and protected with a cryptographic mechanism using a different key known to each recipient.

Thus, if it is desired to broadcast information to all the members of an information domain, group multicasts are likely to be sufficient for most purposes since the member addresses are known. The only real limitation on broadcast communications is that the inherent broadcast capabilities of some media cannot be used.

This page intentionally left blank.

7.0 ADMINISTRATIVE AND ENVIRONMENTAL SECURITY

Reliance on people (i.e., administrative procedures) and the environment is an integral part of achieving total security for an information system. When products are designed and deployed in information systems, administrative and environmental conditions of their use must be met to complement the protection afforded by any hardware and software security mechanisms employed in those products. The specification of such conditions for the use of a component, facility or system is referred to as *security doctrine* in some communities. The administrative and environmental security conditions of use specify how security requirements are to be met and as such are elements of a specific security architecture. As with any design aspect of a specific security architecture, there will be different types of administrative and environmental security allocations, each with different degrees of specificity, which eventually lead to the satisfaction of the required security services through the choice of appropriate security mechanisms.[1] In the case of administrative and environmental security, security services are provided by physical, administrative, personnel, and operational security mechanisms. The DGSA suggests certain security services that can be achieved by administrative and environmental security mechanisms. The designer of more specific security architectures will need to make these, as well as more refined, choices regarding the security service allocations and types of security mechanisms. All, some, or none of the responsibility for provision of each of the security services may be allocated to administrative and environmental security mechanisms. In this section, the allocation suggestions for security services are presented and examples of administrative and environmental security mechanisms that are permissible and consistent with the DGSA are provided.

7.1 ADMINISTRATIVE AND ENVIRONMENTAL SECURITY SERVICE ALLOCATIONS AND MECHANISMS

The DGSA includes availability among the security services. In Section 3, only availability is allocated to the LCS in an LSE, while all the security services are allocated to the environment and to the end systems and relay systems. Environmental mechanisms are expected to protect the end systems, relay systems, and the LCS. Security services implemented in an LSE may take the form of physical, personnel, and administrative security mechanisms. In addition, some types of physical security mechanisms may be incorporated into the hardware of components within an LSE. The definitions of the security services of ISO 7498-2 are extended for use in the DGSA beyond only communications.

An LSE and its components must satisfy the requirements of each of the information domain security policies for which it is accredited. The administrative and environmental security mechanisms employed may vary among information domains. For example, one information domain may require authentication of the identity of an individual through cryptographic based

---

1 Mechanisms, as used here, encompasses manual procedures and physical controls, as well as automated controls.

mechanisms, while another may rely on the simple possession of a badge. An LSE is the principal location for direct implementation of administrative and environmental security mechanisms, but local security mechanisms may also rely upon remote systems to provide initial capabilities and life-cycle support (e.g., key management systems, personnel investigations, shrink-wrapped software, security inspection and testing, security training and awareness).

7.1.1 Mechanisms for Identification and Authentication

Authentication of the claimed identities of individuals, as individuals or as members of a group, is a typical security policy requirement. Authentication mechanisms provide varying degrees of credibility that such claims are correct. Authentication responsibilities are often shared between administrative, environmental, and technical (i.e., hardware and software) mechanisms. Probably the most common mechanism is the picture badge and the guard. The picture on the badge matching the appearance of the holder affirms the association of the individual with what the badge represents. The identity of the individual is thereby authenticated and, in some cases, the possession of the badge establishes further claims. The reading of the magnetic code on a badge matched with the entry of a personal identification number is similar in capability to picture confirmation. Similarly, the matching of fingerprints or retina images authenticates the identity of an individual.

The use of keys with locks, passwords, or cipher lock codes authenticates identity only to the extent of the probability that the presenter is a valid holder of the object or information. That probability is based on the administrative handling and physical protection of such mechanisms or information. The same considerations apply to the use of smart cards, cryptographic ignition keys, and other credentials that make no positive connection with the holder. In general, non-forgeable information bound to the holder is the strongest type of authentication mechanism. Security mechanisms for authentication depend upon system security administrators who perform the initial assignment of the badge or other credential to an individual.

7.1.2 Mechanisms for Access Control

Access control mechanisms enforce security policy requirements for the isolation of assets and information from people and their agents. Access control mechanisms also permit authorized access to assets and information. The first line of protection for the LSE is through mechanisms that control access to the facilities (e.g., buildings, rooms) containing the end systems, relay systems, and LCSs. The human security guard is one of the most familiar types of access control mechanisms. Key, combination, and cypher locks are common mechanisms for controlling access to facilities. Placing an entire LSE within a vault is an extreme form of facility control. With the assumption that only authorized people are in the LSE, surveillance of their activities by security administrators or by co-workers can form the next line of protection. Areas may be declared to require at least two people to be present when activities are in progress ("no-lone" zones).

The next line of protection involves the use of approved containers (e.g., combination safes and locking cabinets) for the protection of system assets. Such containers can be used to protect entire system components (end systems, relay systems, and LCSs) or information storage media (e.g., disks, tapes). Finally, the components themselves may contain access control mechanisms such as power locks, two-person-control devices, and sealed housings.

Within and beyond these lines of protection, access control becomes the responsibility of hardware and software features of the end systems and relay systems. Access control mechanisms can also contribute to the provision of confidentiality, integrity, and availability services; independent aspects of these services are presented in the following sections.

7.1.3 Mechanisms for Confidentiality

Confidentiality mechanisms satisfy security policy requirements to protect information from unauthorized disclosure. The major applications of administrative and environmental confidentiality mechanisms in LSEs involve video displays, printing devices, sounds, and non-video electromagnetic emanations.

Users and security administrators can control when, where, and in whose presence video information is displayed. Video display emanations can be controlled through screen filters and shielded enclosures. Printer ribbon handling, copy counting, and labeling requirements can be controlled by users, operators, and system administrators. The control of trash and the destruction of paper and other media are important procedures. Paper shredders may be useful. Procedures for handling and mechanisms for erasure of persistent storage media can be critical to confidentiality. Sound insulation and sound masking can be used to control disclosure through conversations and machine noises. Electromagnetic emanations, either radiated or conducted, can be confined by shielding rooms and by filtering signal and power wiring using standard TEMPEST features. The presence of copiers and photographic equipment in LSEs requires careful control. Paper and other media devices should be properly wrapped prior to shipping or mailing.

7.1.4 Mechanisms for Integrity

Integrity mechanisms are used in response to security policy requirements to protect information and other system assets from unauthorized modification. The major applications of administrative and environmental integrity mechanisms in LSEs involve the correctness of end system and relay system hardware and software, and the correct functioning and use of other administrative and environmental security mechanisms. System components may have features that permit security diagnostic checking of hardware (for example, through comparison of diagnostic known-answer tests with off-line security check mechanisms). Non-forgeable seals and protective coatings may be used on hardware components and subcomponents to detect or prevent alteration. Cryptographic and non-cryptographic check value mechanisms can be used to ensure the integrity of software packages as delivered and as used.

Regular inspections of facilities and system components is an important part of using integrity mechanisms. Devices used for integrity checking must be stored in protected areas. Software master copies and small system components must also be stored in protected areas while not in use. Protection from electromagnetic interference can be accomplished by filtering and shielding.

7.1.5 Mechanisms for Non-Repudiation

Non-repudiation mechanisms support security policy requirements for proof of delivery and proof of origin of information transactions. Non-repudiation mechanisms may include the contents of a transaction. For paper transactions, notary services and personal signatures are useful mechanisms in providing non-repudiation services. Non-repudiation mechanisms, such as hash coding of data and digital signatures, can be used to validate the source of software packages. Non-repudiation mechanisms could be used for verifying that hardware is unchanged from its manufactured state.

7.1.6 Mechanisms for Availability

Availability mechanisms in communications networks and LSEs satisfy security policy requirements for availability of communications and processing resources. The ability of communications networks to provide timely and regular service depends upon the total security architecture, implementation, and management of those systems. The techniques of redundancy, diversity, contingency reserves, and contingency planning play a large part in communications network availability. Within LSEs, the LCS must be similarly designed and protected to avoid failure outages. Generally, the physical protection and integrity checking of the end systems, relay systems, and LCSs will provide for their availability.

7.2 COTS PRODUCT CONSIDERATIONS

Current COTS products may lack built-in security mechanisms such as those presented in the previous section. Therefore, additional procedures may be required or separate COTS tools that provide a measure of security assurance. COTS products may also be vulnerable to component modification and substitution. Any user not being closely observed may be able to modify or substitute COTS product components to their own benefit or the detriment of the organization. The administrative and environmental mechanisms must ensure that COTS products can be physically accessed only by persons authorized for access to all information in the component unless escorted by someone who is so authorized. At the other extreme, when sufficient built-in isolation mechanisms exist (in GOTS products or custom-designed products), then all communities of interest can be satisfied that physical access is permissible by persons authorized in only one information domain of all those supported.

7.3 SECURITY MANAGEMENT

Security management, as presented in Section 5, includes security service management, security mechanism management, and the management of all security aspects of the system. All of these functions are performed within an LSE. The information domain security manager is an administrator who is authorized to perform installation and maintenance of the information domain security policy representation, access control lists, and other items of the SMIB, such as cryptographic keys. The security manager is provided tools, such as a SMAP, to perform these tasks. The security manager is ultimately responsible for checking personnel clearances, monitoring guard activities, performing audits of security-relevant records, and, in general, supporting all other security mechanisms.

The security aspects of system management are no different from any other applications which require protection. The system must have a security policy and administrative and environmental security mechanisms will be used in support of system management activities. A critical aspect of security management is the training of security administrators and users so that they understand their responsibilities as part of the entire security posture.

This page intentionally left blank.

8.0 EXAMPLE OF A HIGH-LEVEL ARCHITECTURE BASED ON THE DGSA

This section presents an example of how the DGSA's concepts work together and how the DGSA could be used in a typical networked environment. This example is based on a Group Medical Practice (GMP). A GMP was selected to provide an example with which most readers would be familiar and a sufficiently rich environment to demonstrate the concepts of the DGSA. A more detailed example, included in the DGSA Version 1.0, is separately available in "Detailed DGSA Example: Drug Enforcement." Note to demonstrate the concepts of the DGSA, specific detail is provided where necessary. In an actual GMP, additional functions and types of information would be used and additional relationships would exist with internal and external organizations. A number of assumptions are made in this example to facilitate the demonstration of DGSA concepts. These assumptions do not necessarily reflect the operation of an actual GMP, and the reader is cautioned that certain assumptions may invalidate the example in specific legal jurisdictions.

8.1 MISSION

The first stage in developing an information system security architecture is to understand the missions of the organization using the information system. As discussed in Section 2, every organization has missions or goals. For this example, the mission of the GMP is to provide quality health care at a reasonable cost. Most organizations are divided into components, each with its own mission that support the overall mission. Some components of the GMP are the care providers, business office, and laboratories. The care provider's mission is to treat patients according to the principles of the medical profession. The business office's mission is to manage the financial activities of the GMP. The laboratories' mission is to perform medical tests accurately.

8.2 POLICY

Once the GMP mission is determined, the organization must develop a security policy for that mission. The security policy should include requirements from a variety of sources, such as laws and corporate directives. For the GMP example, federal and state laws on privacy require the protection of patient information including the patient's medical, financial, and personal information. Corporate directives define methods of protecting the personnel data on the GMP's care providers and laboratory workers. For example, the GMP's security policy states that only the personnel department, the supervisor, and the employee may access an employee's personnel folder.

Another source of requirements for the GMP security policy is the perceived threat environment. Threats can be internal or external. An example of an internal threat is the embezzlement of GMP funds. An example of an external threat is a tabloid attempting to access a patient's medical history. For the GMP example, the threats are primarily aimed at the integrity and confidentiality of the GMP's information objects. This threat environment leads to requirements for high strength of service for identification and authentication (I&A), confidentiality, and

integrity. These requirements, combined with the requirements derived from the laws and corporate directives, generate the GMP security policy. The GMP security policy serves as the common basis for the development of security policies for each of the information domains.

8.3 INFORMATION DOMAINS

An information domain as defined in Section 3 is a set of users, their information objects, and a security policy. The security policy for the GMP identifies information domains and their constituent elements. These information domains are tied directly to the missions that they support. Some of the GMP's information domains are the patient medical history, patient financial information, laboratory records, accounting, and patient address information. Each of these information domains supports one or more of the GMP's missions. For example, the patient address information domain is constructed to support the care providing and business office missions. While the GMP example uses a number of information domains, only the patient medical history information domain is presented in depth. A patient medical history information domain is created for each patient in the GMP. For this example, it is assumed that all patients have a primary medical care provider or doctor.

The set of users of the patient medical history information domain includes the patient, the patient's doctors, their nurses, and the medical director of the GMP. This membership limits the access to a patient's medical history to only those individuals directly involved with the patient. The medical director has access for emergency situations and for internal situations in the GMP. Membership in the information domain is not static. Staff turnover or the need for consultation by a specialist will cause changes in the membership of the information domain. The patient's primary doctor has the authority to modify the membership of this information domain.

Examples of information objects within the patient medical history information domain include test results, prescriptions, and reports on a patient's medical visits. Each of these information objects is uniquely identifiable and directly associated with its information domain. In addition, the GMP requires protection of each information object in the patient medical history information domain to ensure the integrity and authenticity of the data.

The final, and perhaps the most critical, element of the information domain is the information domain security policy. The information domain security policy comprises the roles and privileges of the members and the protections that must be applied to the information objects within the information domain and the transfer policy. The transfer policy addresses inter-domain and intra-domain transfers of the information objects. The information domain security policy identifies the security services required for operation within an information domain. Each security service has a strength of service characteristic. For the GMP example, the value of the strength of service is specified as a low, medium or high level of assurance.

The roles, privileges, and protections of the patient medical history information domain security policy are:

- Membership in the patient medical history information domain includes the patients, their doctors and nurses, and the medical director.
- All members of the information domain must be identified and authenticated at a high level of assurance.
- Every member is allowed to view the information objects in the information domain.
- Members cannot modify the contents of any of the existing information objects.
- The deletion of information objects in the information domain requires the consent of both the patient and the doctor of record.
- The integrity of the information objects in the information domain must be protected at a high level of assurance.
- The confidentiality of the information objects in the information domain must be protected at a high level of assurance.
- The identity of the creator of an information object must be protected. Therefore, non-repudiation of origin of an information object in this information domain must be protected at a high level of assurance.
- The availability of the information objects of the patient medical history information domain is at a moderate level of assurance.

The transfer policy for the patient medical history information domain is:

- The confidentiality of all information objects must be maintained during inter- and intra-domain transfers at a high level of assurance.
- All outgoing inter-domain transfers of information objects must be approved by the patient before the data can be transferred to another information domain.
- In an emergency, such as patient incapacitation, the medical director is authorized to release the patient's medical information to a physician treating the patient who is not already a member of the information domain.
- All incoming inter-domain transfers of information objects are accepted, if the integrity of the information objects is verifiable and if they pertain to the patient.

The described patient medical history information domain security policy may not address every issue of a patient's medical records, but serves as an example of the type of material in an information domain security policy. This material is derived from the mission requirements and

the GMP security policy. It should be noted that, while there must be a patient medical history information domain for every GMP patient, the same information domain security policy can be used. If exceptional circumstances arise, the basic information domain security policy can be modified.

Section 8.5 presents three scenarios to demonstrate the concept of operations for a DGSA based architecture. These scenarios require a variety of information domains to demonstrate the concepts of the DGSA. Figure 8-1 lists the information domains used in each scenario.

8.4 INFORMATION SYSTEM SECURITY ARCHITECTURE

This section presents a GMP information system security architecture based on the DGSA concepts. In this system, end systems are available to all employees of the GMP (e.g., doctors, nurses, or administrative staff), all patients, and all organizations (e.g., hospitals, insurance firms). Access rights to the respective information domains vary depending on the role of the GMP employee or the association with the patient (e.g., the patient's primary physician, hospital nurse).

New Patient Enrollment	Medical Visit	Hospital Admission
Patient Address Information(address, phone) Domain	Patient Address Information Domain	Patient Address Information Domain
Patient Financial Information Domain	Child Medical History Information Domain	Hospital Billing and Insurance Information Domain
Patient Medical History Information Domain	Patient Medical History Information Domain	Patient Medical History Information Domain
Doctor Appointment Calendar Information Domain	Lab Appointment Calendar Information Domain	Hospital Patient Medical History Information Domain
Accounting Information Domain	GMP Lab Information Domain	Hospital Lab Information Domain
Security Management Information Domain	Security Management Information Domain	Security Management Information Domain

Figure 8-1. Information Domains for the Scenarios

Figure 8-2 depicts an example GMP architecture. Although a GMP may have other end systems for additional components, only the four end systems shown below are specifically addressed in the example scenarios. For each of the four end systems discussed, the end system security policy, functionality, and security service allocations are identified based on the information domains that are supported by those end systems. For an information domain to be supported on an end system, the end system must be capable of implementing the information domain security policy. For this example, a single security management information domain is maintained across all end systems and information domains of the GMP in accordance with the one to many paradigm described in Section 5. The security management information domain contains the information domain security policies and other security critical information objects.

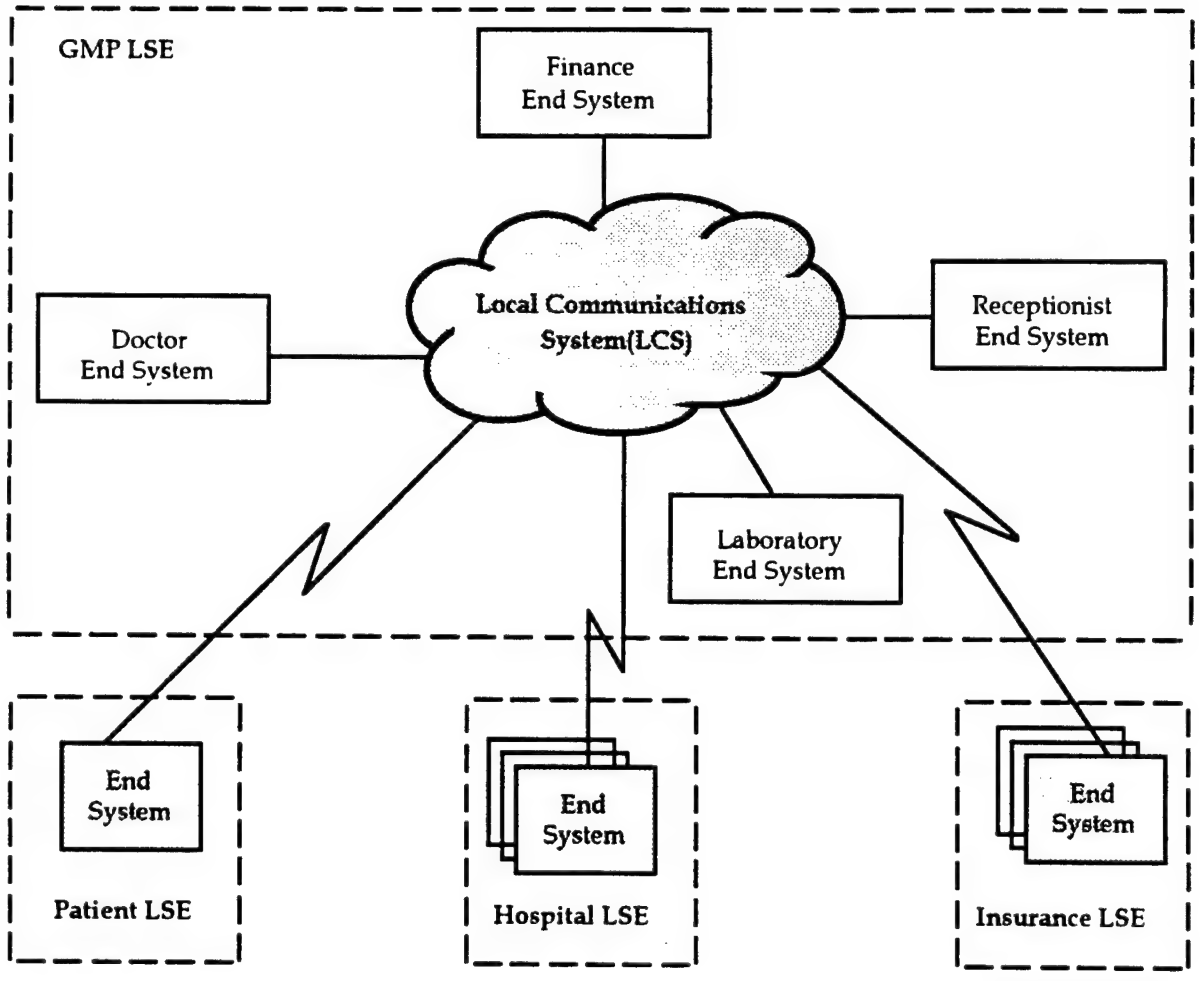

Figure 8-2. Architecture for GMP Example

The receptionist end system is used to schedule and check appointments and maintain calendars for the doctors of the GMP. This system is used to maintain records of patient identification information, such as address and telephone number. The receptionist end system provides support for the patient address information domain and the appointment calendar information domain. The receptionist end system must provide strict isolation at a moderate level of assurance to support the information domains that are resident on this end system. The receptionist end system must provide confidentiality and I&A security services at a medium level of assurance to ensure that patient address and doctors' calendar information is not released outside the membership of the respective information domains. A low level of assurance is required for the security services of integrity, non-repudiation, and availability.

The laboratory staff uses the lab end system to schedule tests, to monitor lab personnel availability, and to record test results. The lab end system provides support for the medical history information domain, patient address information domain, lab results information domain, lab calendar information domain, and GMP security management information domain. The lab end system must provide strict isolation at a high level of assurance to support the information domains that are resident on this end system. The lab end system must provide confidentiality, integrity, I&A, and non-repudiation security services at a high level of assurance. These services ensure that medical history and lab result information is not released outside the membership of the respective information domains, patient privacy is protected, and medical history and test results cannot be altered. The requirement for availability security services is a medium level of assurance.

The doctor end system is used by a doctor of the GMP to record and update patient medical history records. Although a patient may request a copy of the records for their own end system, the doctor's version of these records is the master copy. The doctor end system provides support for the medical history information domain, patient address information domain, doctors' calendar information domain, and GMP security management information domain. This end system must provide strict isolation at a high level of assurance to support the information domains that are resident on this end system. This end system must provide confidentiality, integrity, I&A, and non-repudiation security services at a high level of assurance. These services ensure that medical history information is not released outside the membership of that information domain, patient privacy is protected, and patient medical histories cannot be altered. The requirement for availability security services is a medium level of assurance.

The finance end system is used by the financial staff of the GMP to record patient insurance information, patient billing information, and insurance billing and payments. The finance end system provides support for the patient address information domain, patient financial information domain, accounting information domain, and the GMP security management information domain. This end system must provide strict isolation at a moderate level of assurance to support the information domains that are resident on this end system. This end system must provide integrity and identification and authentication security services at a medium level of assurance. These services ensure that the GMP's financial information and billing information is not released outside the membership of the finance information domain and cannot be altered or deleted. The policy requires confidentiality and non-repudiation security

services at a medium level of assurance to ensure that adequate protection of patient financial information. The requirement for availability security services is low, since the system need only be available when the financial office is open (e.g., Monday through Friday, 8:30 AM to 5:30 PM).

The security policy for the security management information domain of the GMP end systems indicates that security mechanisms must be available on all end systems to support the establishment of information domains. For example, mechanisms are required to create the memberships and to install the security policies of the various information domains.

In addition to the security service allocations and strict isolation requirements identified above, the overall GMP LSE security policy requires that the LCS provide availability security services at a medium level of assurance to ensure that all GMP end systems are able to communicate as needed when the GMP is open. The LCS is also required to provide confidentiality and integrity security services to protect the information in transmission within the GMP at a medium level of assurance. The GMP LSE security policy requires that administrative and environmental controls at a medium level of assurance be in place to safeguard physical access to all GMP end systems.

Figure 8-3 provides a mapping between the requirements identified for the medical history information domain and the security service allocations across the end systems of the GMP. All requirements of the medical history information domain are addressed by the allocation of security services to the doctor and lab end systems. On the bases of the allocations of security services, security mechanisms can be chosen to provide the requisite strength of service.

The Hospital LSE is composed of a collection of end systems that serve different purposes (e.g., financial, patient check-in). This collection of end systems is treated as a single end system here to simplify this example. The hospital end system communicates with the GMP doctor end system. The Hospital Patient Medical History information domain is created on the GMP doctor end system.

The Insurance LSE, like the Hospital LSE, is actually a collection of end systems that serve different purposes (e.g., billing receipt, requests for insurance information, insurance claims, payments made). This collection is treated as a single insurance end system in this example for simplicity. The insurance end system communicates with the GMP finance end system.

The patient end system may be used to communicate with:

- The receptionist end system of the GMP to establish appointments
- The hospital end system to establish hospital test or lab appointments
- The insurance end system to identify any errors or to present the issues of a specific case.

Each patient is assumed to have an end system readily available to them.

Medical History Information Domain Requirements	Security Service Allocation Within the GMP
Each information object is digitally signed to prevent modification	High assurance integrity security services for the doctor and lab end systems
Membership limited to patient, doctor, nurses, and medical director and strong I&A is applied	High assurance identification and authentication security services for the doctor and lab end systems
Objects only deleted by joint permission of patient and doctor	High assurance integrity security services for the doctor and lab end systems
Integrity of patient medical information must be maintained	High assurance integrity security services for the doctor and lab end systems
Confidentiality of patient medical information must be maintained	High assurance confidentiality security services for the doctor and lab end systems
Information object creator must be identifiable	High assurance non-repudiation security services for the doctor and lab end systems
Patient medical histories must be reasonably available	Medium assurance availability security services for the LCS and ESs
Confidentiality must be maintained during any information transfers	High assurance confidentiality security services for the doctor and lab end systems
Outbound inter-domain transfers must be approved by the patient	High assurance access control security services for the doctor and lab end systems
Medical director can release medical information in an emergency	High assurance access control security services for the doctor and lab end systems
Incoming inter-domain transfers must be verifiable and pertinent to the patient	High assurance integrity and non-repudiation security services for the doctor and lab end systems

Figure 8-3. Mapping of Requirements to Security Service Allocations

8.5 SCENARIOS

This section presents three scenarios for the GMP example. These scenarios demonstrate:

- The creation and instantiation of information domains
- Creation of information objects
- Creation and use of security contexts and security associations
- Use and establishment of access privileges
- Transfer of information objects between information domains (inter-domain and intra-domain)
- Creation and use of multidomain objects
- Switching from one information domain to another.

For all three scenarios, the doctor and patient jointly control the information in the patient's medical history information domain. Doctors may create new information objects in this information domain and read any existing information objects in this information domain. When an information object is created, it must be signed (using a digital signature that is public key based) to protect its integrity. That is, once a medical history information object has been created for a patient it must not be altered. Patient medical history information objects must not be deleted without the consent of both the doctor and the patient. A patient may obtain a copy of any of his medical history information objects. The copy of the medical history information object retains the digital signature of the originator and therefore cannot be modified without being detected.

A doctor can transfer copies (e.g., transfer specific records) to different information domains, for example to a hospital information domain, with the consent of the patient. Medical specialists can become members of the information domain on a temporary basis if the doctor and patient both agree to permit the specialist to access the patient information. Alternatively, a temporary information domain containing copies of only the pertinent medical history information can be created with the doctor, the patient, and the specialist as its members.

Normally, a hospital obtains a copy of selected information objects, as necessary, from the doctor via an information transfer. The hospital has one information domain per patient and appropriate hospital employees have read access to all information in that information domain. Appropriate hospital staff may also add new information objects to the information domain, as necessary. A transfer policy permits information to be sent between a hospital information domain and the patient's medical history information domain. When new objects are introduced into the hospital information domain, a copy of the object is immediately transferred into the patient's medical history information domain. All objects in the hospital information domain carry a digital signature and cannot be modified by the patient.

Patient information is dispersed among several information domains. Multidomain objects are created to simplify the presentation of patient information. For example, the finance information domain contains all patient billing and insurance related information objects, and the personal address information domain contains all identification information for the patient. A multidomain object is created which links these information objects to facilitate the operation of the finance mission. These information objects provide a look-at-a-glance for all GMP financial staff regarding patient information. Similar links are created among other information domains in the GMP but are not described further in this example.

8.5.1 Scenario 1: New Patient Enrollment

A request for an appointment is sent from the patient's end system to the receptionist end system at the GMP. The receptionist schedules an appointment based on the patient's name and sends an acknowledging message to the patient's end system.

Upon arrival at the GMP for the first appointment, the receptionist creates an information object for the patient in the GMP patient address information domain. The information object contains identification information, such as patient name, address, home telephone number, and work telephone number. The identification information is then transferred from the receptionist end system to the finance end system based on the intra-domain transfer policy for the patient address information domain.

The new patient then speaks with a financial staff representative who obtains additional information, regarding responsibility for bills and insurance coverage. A financial information object is created in the finance information domain using that information. The financial staff representative establishes a multidomain object for the patient in the accounting information domain. That information object points to the patient address information object in the patient address information domain and the finance information object in the financial information domain.

The doctor's nurse creates a medical history information domain for the new patient and obtains a medical history from the patient. The medical history is recorded as information objects in the new medical history information domain. From this point forward, access to this new medical history information domain requires user authentication.

At the first doctor/patient meeting the patient's medical history is reviewed. The doctor may update the patient's medical history information objects after their initial consultation. After reviewing the patient's history, the doctor uses a digital signature to sign the information objects so that they may no longer be altered. The doctor creates new information objects in the medical history information domain to record the events of this appointment. If the doctor requires a follow-up appointment or lab tests, a message is sent to the receptionist requesting that such appointments be scheduled before the patient leaves.

If necessary, the GMP receptionist updates the doctor's appointment calendar by establishing another appointment with the patient. The receptionist end system transfers a copy of the patient address information for this patient to the patient address information domain on the lab end

system and requests test scheduling. The lab end system creates a new object for this patient in the lab test information domain and establishes an appointment. Later, a multidomain information object is created that contains pointers to the patient identification information and the patient test results information.

8.5.2 Scenario 2: Medical Visit

This scenario builds upon the new patient enrollment scenario. The patient makes an appointment with his or her doctor through the receptionist, as described in the previous scenario. When the patient visits the doctor, the doctor first authenticates himself to the end system in order to accesses the patient's medical information to review the patient's status.

The patient is suffering from a minor ailment, but as a preventive measure, the doctor orders lab tests. Until the laboratory results have been completed, the doctor issues an interim prescription to alleviate the patient's ailment. The doctor creates the prescription on his end system and sends a copy of the prescription electronically to the patient's pharmacist. The doctor digitally signs the prescription, so that the pharmacist can verify its integrity and authenticity. The prescription is encrypted, in accordance with the patient's privacy requirements, during transmission through the network. This transfer is accomplished by accessing the pharmacist's certificate stored in the public key certificate directory and using the pharmacist's public key for encryption.

During this visit, the patient asks a question about the results of tests done for the patient's child. It is assumed for this scenario that the patient is the child's legal guardian and that the doctor is the primary care provider for the child. The doctor attempts to access the child's medical information. Since the end system has previously authenticated the doctor, the end system must only determine whether the doctor is a member of the child's medical history information domain. After verifying that the doctor is a member of that information domain, the system grants access to the data and the doctor is able to answer the parent's question. This scenario assumes that the child's information is available on this end system. If the information is stored on a different end system then the doctor's end system must make a connection to another end system in the GMP, such as a database server. The doctor then accesses the information directly on the server or the information object is transferred to the doctor's end system. In any case, the end system must ensure that the confidentiality of the information is protected while it moves through the GMP local communication system. There is a high assurance requirement for the confidentiality of this information. Note this requirement may have been satisfied through extensive environmental and administrative procedures used by the GMP to protect its local subscriber environment and a simple cryptographic mechanism.

After the patient's visit has ended, the doctor completes a report for the visit which becomes a new information object within the patient's medical history information domain after the doctor has digitally signed it. The doctor sends a statement to the financial office so that the patient's insurance company can be billed for the routine medical visit. Upon receipt of the doctor's statement, the financial office creates a bill within the financial information domain. Since the

GMP previously established a transfer policy with the insurance company, the bill is transferred to the insurance company in accordance with this policy.

8.5.3 Scenario 3: Hospital Admission

The results of the laboratory tests conducted during the patient's visit indicate a more serious medical problem that requires a short stay in the hospital. The doctor creates a tentative transfer request for the patient medical information that is needed by the hospital. The doctor arranges for the patient to return for an office visit to discuss the results of the laboratory tests. If the patient agrees on the need for the hospital stay, the patient must give electronic consent before the doctor's end system releases the patient's medical information to the hospital. After the patient is authenticated by the doctor's end system, the patient reviews the transfer request, and, assuming concurrence, the patient digitally signs the transfer request.

The doctor, as a member of the staff of the hospital, makes arrangements for the patient to enter the hospital. The hospital creates a medical history information domain for the patient on the hospital end system. The doctor then initiates the transfer of the patient's medical information. The security policy enforcement function on the doctor's end system checks to see that both the doctor and the patient consented to the transfer before releasing the data to the hospital's patient medical history information domain. (In an emergency, this information could be released by the medical director of the GMP without the patient's consent.)

The actual transfer is accomplished by creating the patient's hospital medical history information domain on the doctor's end system (in accordance with prior agreements between the hospital and the GMP). An application on the doctor's end system causes the creation of security contexts for both the GMP and hospital patient medical record information domains and the previously approved interdomain transfer takes place. The transfer of information to the hospital end system requires the establishment of a security association between the doctor's end system and the hospital's end system over their common communications network. The security association maintains the confidentiality of the information during transfer. The first step in creating the security association is for the doctor's end system to verify that it is connected to the hospital's end system. Once this connection has been confirmed, the end systems security management functions negotiate the parameters of the security association to satisfy the requirements of the hospital patient medical record information domain's transfer policy. Since the communication network only provides the security service of availability, a strong cryptographic mechanism is employed to provide the requisite level of confidentiality. The completion of the negotiation establishes a distributed security context between the two end systems and the secure transfer of the information objects.

During the patient's stay in the hospital, any medical information objects that are created by the hospital are transferred into the GMP's patient medical history information domain. The process is the reverse of that used for the transfer into the hospital's medical history information domain. After the completion of the patient's stay in the hospital, the hospital archives the patient medical history information domain.

APPENDIX A

REFERENCES

Note: References appearing in this section represent documents used in preparation of this volume, including some sources used at the time of initial document development that may no longer be current or applicable. The reader is advised to check the current applicability of a reference appearing in this list before using it as an information source. The reference section will be completely reviewed and revised for the next release of the TAFIM.

1. Abrams, Marshall D. and Michael V. Joyce, January 1993, *On TCB Subsets and Trusted Object Management*, MITRE Technical Report 92W0000248, McLean, VA.

2. Center for Information Systems Security (CISS), 30 January 1995, *Department of Defense Goal Security Architecture Transition Plan*, Defense Information Systems Agency, Washington, DC.

3. Case, Jeff, Mark Fedor, Martin Schoffstall and James Davin, 1989, *Simple Network Management Protocol (SNMP)*, Internet Request for Comments 1098.

4. Case, Jeff, 1991, *SNMP Version 2*, Internet Request for Comments 1441.

5. International Telegraph and Telephone Consultative Committee (CCITT), 1988, *Recommendations X.400-X.420: Data Communications Networks, Message Handling Systems.*

6. _____, 1992, *Recommendations X.500-X.521 Data Communications Networks, Directory.*

7. Department of Defense, 1985, *Department of Defense Trusted Computer System Evaluation Criteria*, DoD 5200.28-STD, Washington, DC.

8. Institute of Electrical and Electronic Engineers (IEEE), *Standard for Interoperable LAN/MAN Security, Part 2--Secure Data Exchange Protocol Specification*, IEEE 802.10b.

9. _____, 1995, IEEE *Standard for Interoperable LAN/MAN Security, Clause 3-Key Management Protocol Specification (Draft)*, IEEE 802.10c.

10. International Organization for Standardization (ISO), 1989a, *Information Processing Systems -- Open Systems Interconnection -- Basic Reference Model -- Part 2: Security Architecture*, ISO 7498-2.

11. _____, 1989b, *Information Processing Systems -- Open Systems Interconnection -- Basic Reference Model -- Part 4: Management Framework*, ISO 7498-4.

12. _____, 1991, *Information Technology -- Common Management Information Protocol -- Part 1: Specification*, ISO/IEC 9596-1.

13. _____, 1994a, *Information Technology -- Open Systems Interconnection -- Basic Reference Model: The Basic Model*, ISO/IEC 7498-1.

14. _____, 1994b, *Information Technology -- Open Systems Interconnection -- General Upper Layers Security -- Part 3: Security Exchange Service Element (SESE) Protocol Specification*, ISO/IEC 11586-3.

15. _____, 1995a, *Information Technology -- Open Systems Interconnection -- Network Layer Security Protocol*, ISO/IEC 11577.

16. _____, 1995b, *Information Technology -- Telecommunications and Information Exchange Between Systems -- Transport Layer Security Protocol*, ISO/IEC 10736.

17. _____, 1995c, Information Technology -- *Open Systems Interconnection - Security Frameworks for Open Systems - Part 1: Overview*, ISO/IEC DIS 10181-1.

18. _____, 1995d, Information Technology -- *Open Systems Interconnection - Security Frameworks for Open Systems - Part 3: Access Control*, ISO/IEC 10181-3.

19. Linn, John, September, 1993, *General Security Services Application Program Interface*, Internet Engineering Task Force, Request for Comments: 1508.

20. National Security Agency, 22 February 1993, Draft *Department of Defense Information Systems Security Policy*, DISSP-SP.1.

21. Rushby, J., September, 1984, A Trusted Computing Base for Embedded Systems, Proceedings of the 7th DOD/NBS Computer Security Symposium, pp. 294-311.

APPENDIX B

ACRONYMS

ADF	Access Control Decision Function
AEF	Access Control Enforcement Function
API	Application Program Interface
C&A	Certification & Accreditation
C4I	Command, Control, Communications, Computers, and Intelligence
C4IFTW	C4I for the Warrior
CCITT	International Telegraph and Telephone Consultative Committee
CIM	Center for Information Management
CISS	Center for Information System Security
CMIP	Common Management Information Protocol
CN	Communications Network
COTS	Commercial-Off-the-Shelf
DGSA	DoD Goal Security Architecture
DIS	Defense Information System
DISA	Defense Information Systems Agency
DISN	Defense Information System Network
DISSP	Defense Information Systems Security Program
DMS	Defense Message System
DNS	Domain Name Service
DoD	Department of Defense
DSA	Directory Service Agents
EKMS	Electronic Key Management System
ES	End System
GMP	Group Medical Practice
GOTS	Government-Off-the-Shelf
GSS	General Security Service
GULS	General Upper Layer Security
I&A	Identification and Authentication
IEEE	Institute of Electrical and Electronics Engineers
IETF	Internet Engineering Task Force
ISO	International Organization for Standardization
ITSDN	Integrated Tactical/Strategic Data Network
LAN	Local Area Network
LCS	Local Communications System

LMD	Local Management Device
LSE	Local Subscriber Environment
MAP	Management Application Process
MAN	Metropolitan Area Network
MIB	Management Information Base
MISSI	Multilevel Information System Security Initiative
MLS	Multilevel Security
MSP	Message Security Protocol
NIC	Network Information Center
NLSP	Network Layer Security Protocol
NSA	National Security Agency
OSD	Office of the Secretary of Defense
OSI	Open Systems Interconnection
RM	Reference Model
RS	Relay System
RVM	Reference Validation Mechanism
SAMP	Security Association Management Protocol
SDE	Secure Data Exchange
SDNS	Secure Data Network System
SESEP	Security Exchange Service Element Protocol
SILS	Secure Interoperable LAN/MAN Standard
SMAP	Security Management Application Process
SMIB	Security Management Information Base
SNMP	Simple Network Management Protocol
SPDF	Security Policy Decision Function
SPEF	Security Policy Enforcement Function
TAFIM	Technical Architecture Framework for Information Management
TFS	Traffic Flow Security
TLSP	Trasport Layer Security Protocol
U.S.	United States
VLSI	Very Large Scale Integration
WWW	World Wide Web